MICRO-CONTROLLER

Was ist ein Micro-Controller ?

Controller = Steuerung

Ein Controller wird zur Steuerung eines physischen Prozesses eingesetzt.

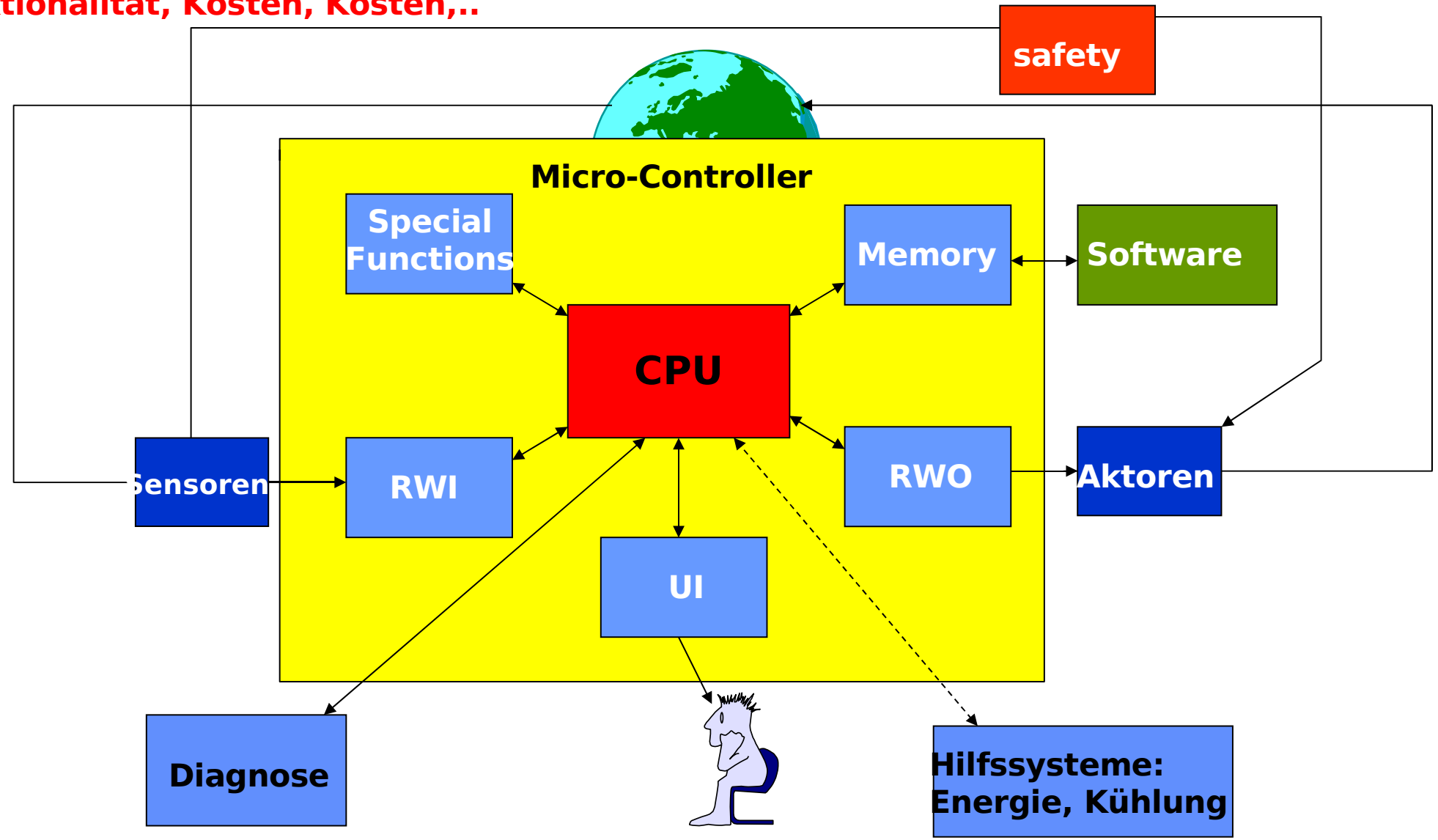
Die Realisierung eines Controllers kann auf viele verschiedene Arten erfolgen, z.B. ein Schaltschrank mit Relais, ein analoger Regelkreis oder eine speziell aufgebaute digitale Logikschaltung.

Speicherprogrammierbare Steuerung (SPS)
Programmable Logic Controller (PLC)

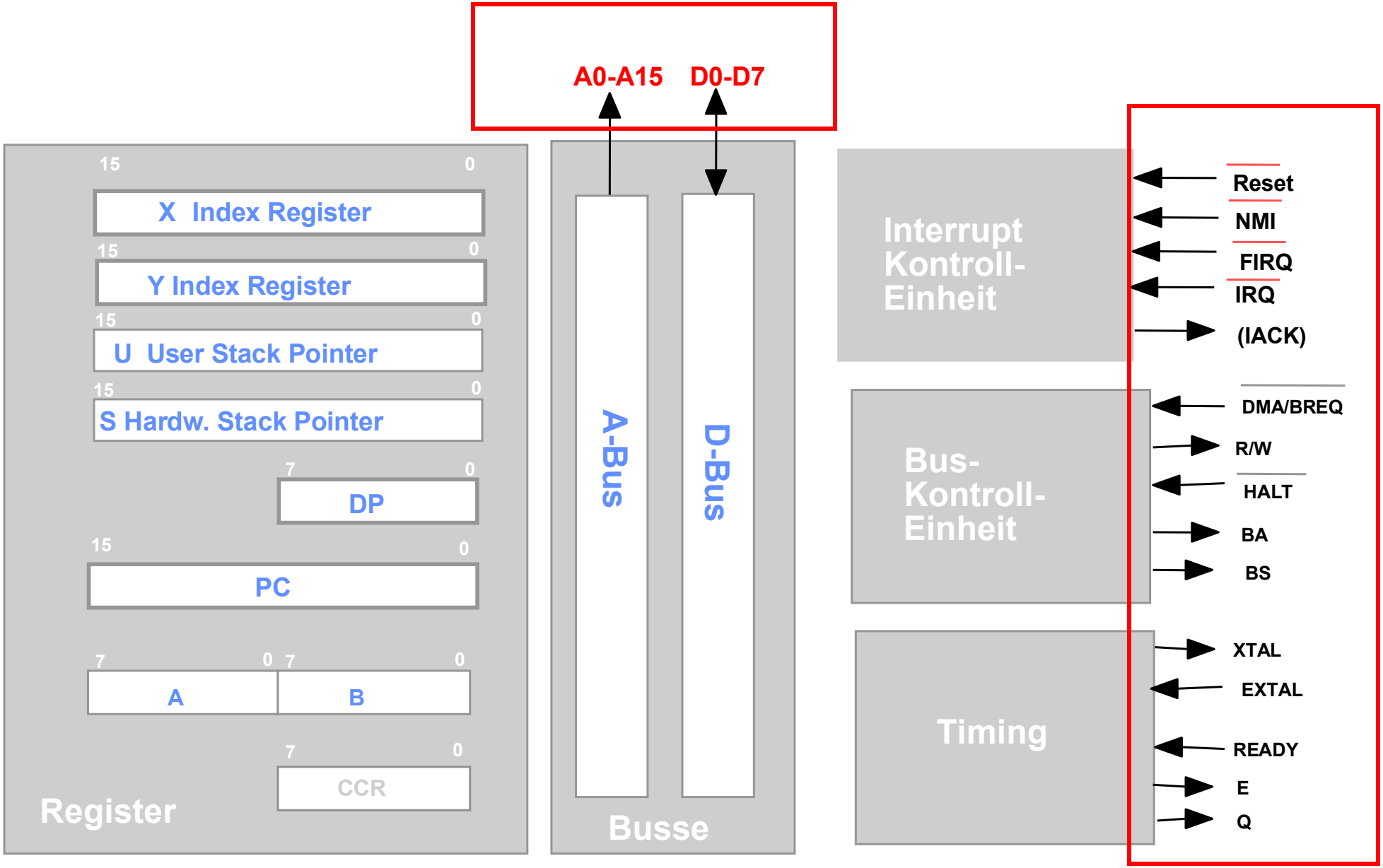
Ein Micro-Controller ist eine Steuerungskomponente, deren Funktionen von einem Microprozessor kontrolliert werden.

stemsicht eines Architekten für Kontrollsysteme:

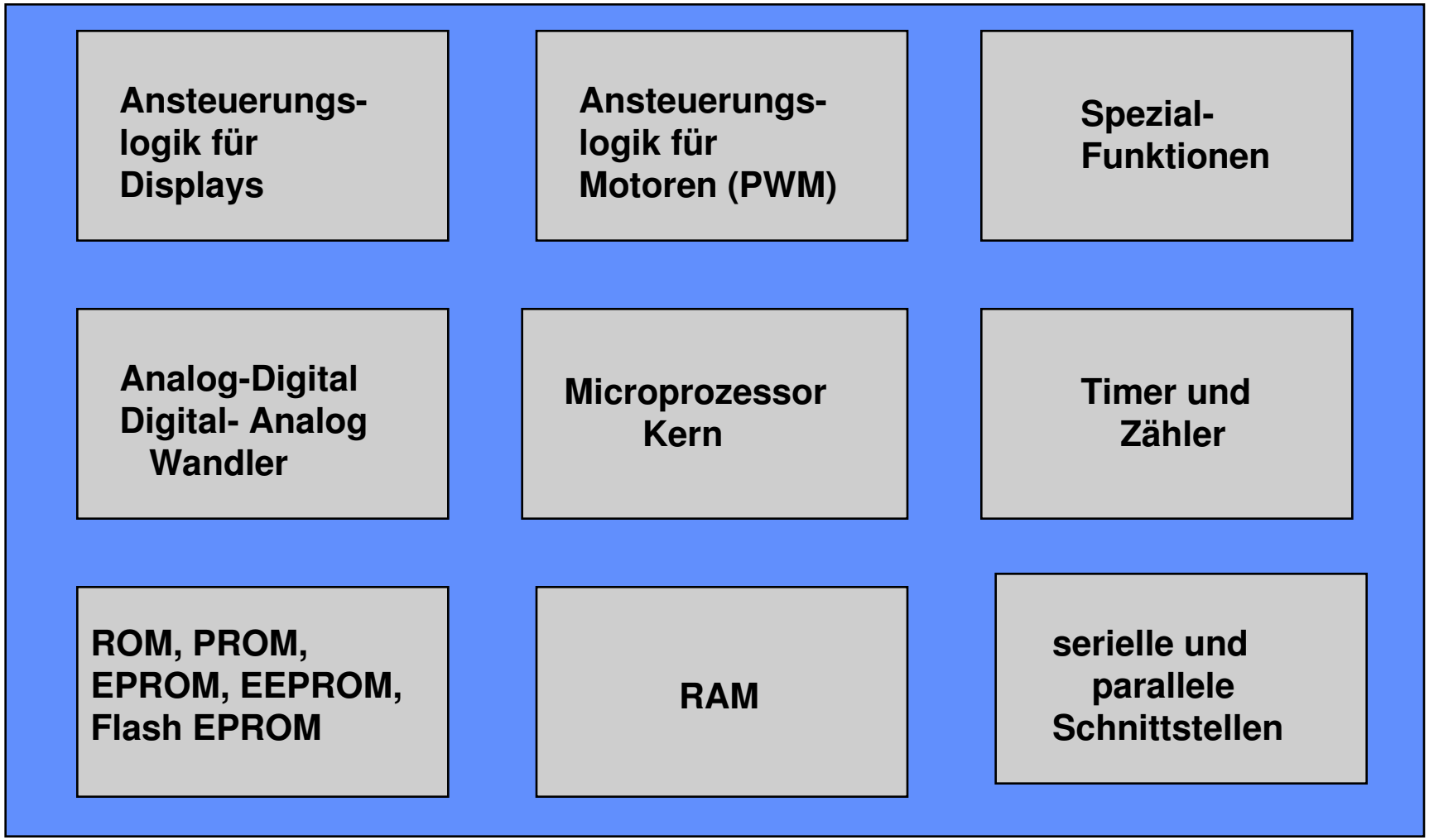
eistungs-Eigenschaften gemessen in: Kosten, Time-to-Market, Kosten, unktionalität, Kosten, Kosten,..



Schnittstelle eines Microprozessors: Speicherbus



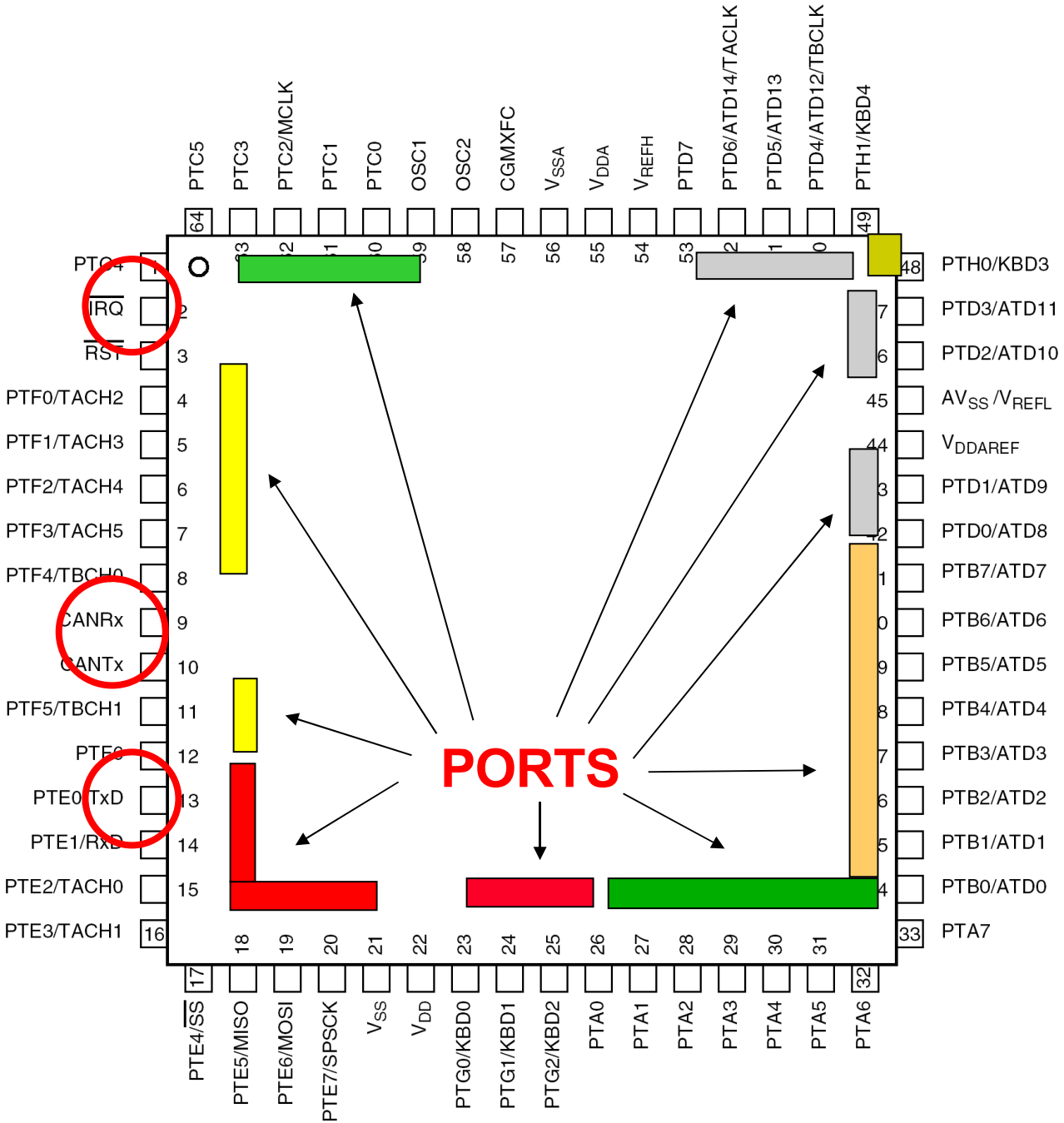
Micro-Controller

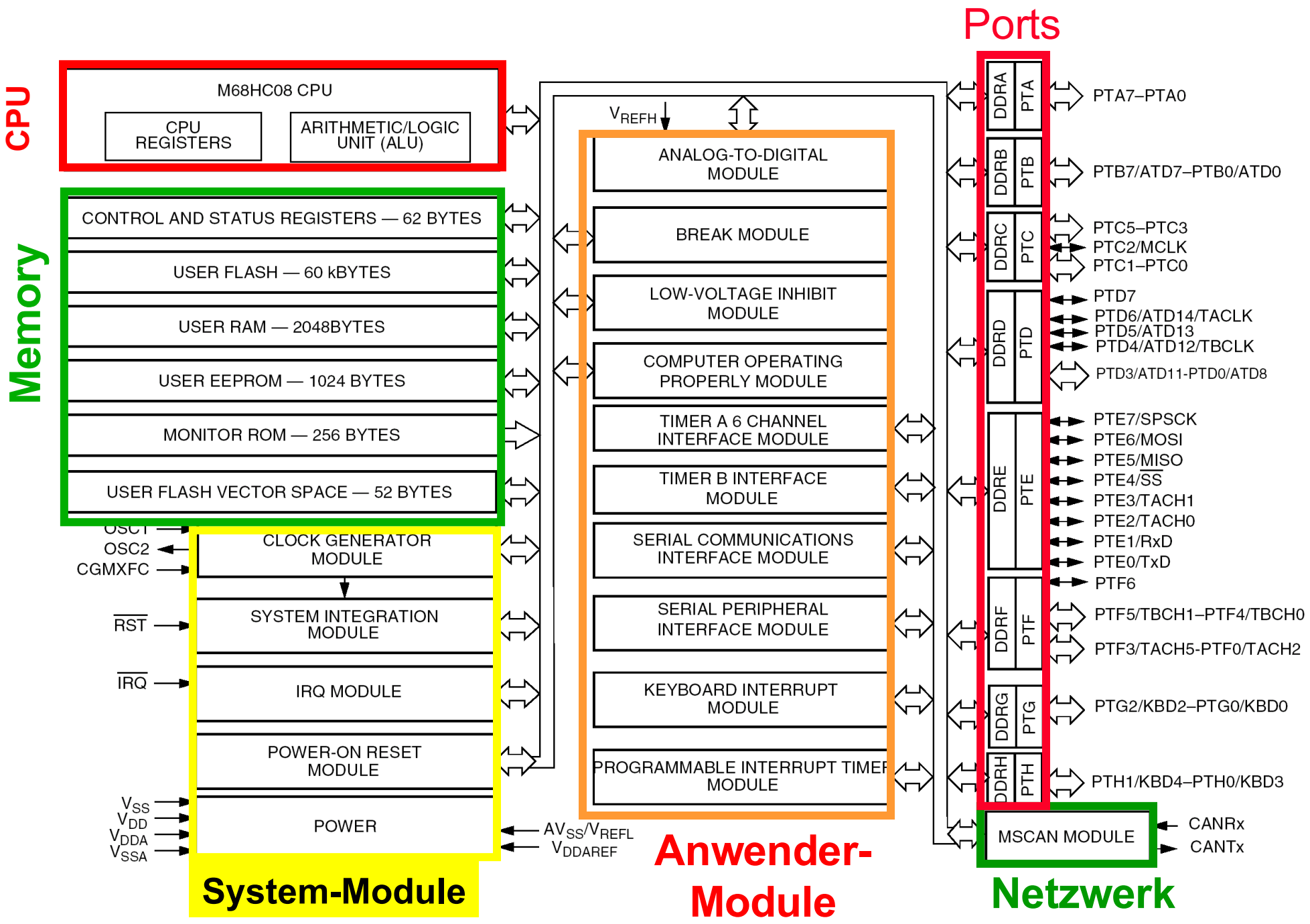


Schnittstelle zu einem Micro-Controller:

PORTS

Beispiel:
Motorola
MC 68HC908AZ60A




CPU
M68HC08 CPU
CPU REGISTERS
ARITHMETIC/LOGIC UNIT (ALU)
Memory
CONTROL AND STATUS REGISTERS — 62 BYTES
USER FLASH — 60 kBYTES
USER RAM — 2048BYTES
USER EEPROM — 1024 BYTES
MONITOR ROM — 256 BYTES
USER FLASH VECTOR SPACE — 52 BYTES
OSC1
OSC2
CGMXFC
CLOCK GENERATOR MODULE
RST
SYSTEM INTEGRATION MODULE
IRQ
IRQ MODULE
POWER-ON RESET MODULE
VSS
VDD
VDDA
VSSA
POWER
System-Module
VREFH
ANALOG-TO-DIGITAL MODULE
BREAK MODULE
LOW-VOLTAGE INHIBIT MODULE
COMPUTER OPERATING PROPERLY MODULE
TIMER A 6 CHANNEL INTERFACE MODULE
TIMER B INTERFACE MODULE
SERIAL COMMUNICATIONS INTERFACE MODULE
SERIAL PERIPHERAL INTERFACE MODULE
KEYBOARD INTERRUPT MODULE
PROGRAMMABLE INTERRUPT TIMER MODULE
AVSS/VREFL
VDDAREF
Anwender-Module
Ports
DDRA
PTA
PTA7–PTA0
DDRB
PTB
PTB7/ATD7–PTB0/ATD0
DDRC
PTC
PTC5–PTC3
PTC2/MCLK
PTC1–PTC0
DDRD
PTD
PTD7
PTD6/ATD14/TACLK
PTD5/ATD13
PTD4/ATD12/TBCLK
PTD3/ATD11-PTD0/ATD8
DDRE
PTE
PTE7/SPSCK
PTE6/MOSI
PTE5/MISO
PTE4/SS
PTE3/TACH1
PTE2/TACH0
PTE1/RxD
PTE0/TxD
PTF6
DDRF
PTF
PTF5/TBCH1–PTF4/TBCH0
PTF3/TACH5-PTF0/TACH2
DDRG
PTG
PTG2/KBD2–PTG0/KBD0
DDRH
PTH
PTH1/KBD4–PTH0/KBD3
MSCAN MODULE
CANRx
CANTx
Netzwerk

Micro-Controller-Familien :

Zielanwendung	Spez. Komponenten	Nutzung
General Purpose	Timer, A/D, EEPROM, Input Capture/Output Compare serielle und synchr.Schnittstellen	logic replacement,
Automotive	EEPROM, CAN A/D, On-Chip Spannungs- stabilisator	Electric Seat Control Klima, Radio, Alarm, IR-Schlüssel Zündung, Air Bag, etc.
Computer	Monitorsteuerung (hor./vert. Sync), PWM	Tastatur-, Maus-, Monitorkontrolle
Consumer	Multitask Support, LCD-Treiber	Waschmaschinen, CD-Spieler, Handy Fernsteuerung
Industrial	EEPROM, A/D, Timer, PWM, CAN	SPS, Motor-Kontrolle, Thermostat
Telecommunications	EEPROM, DTMF-receiver +generator A/D, D/A, Tongenerator	Digitale Übertragung, Handy-Kontrolle
TV + Video	EEPROM, On-Screen-Display-Supp. LCD- und andere Anzeigetreiber	Videorecorder-Kontrolle, Bildschirm-Menues

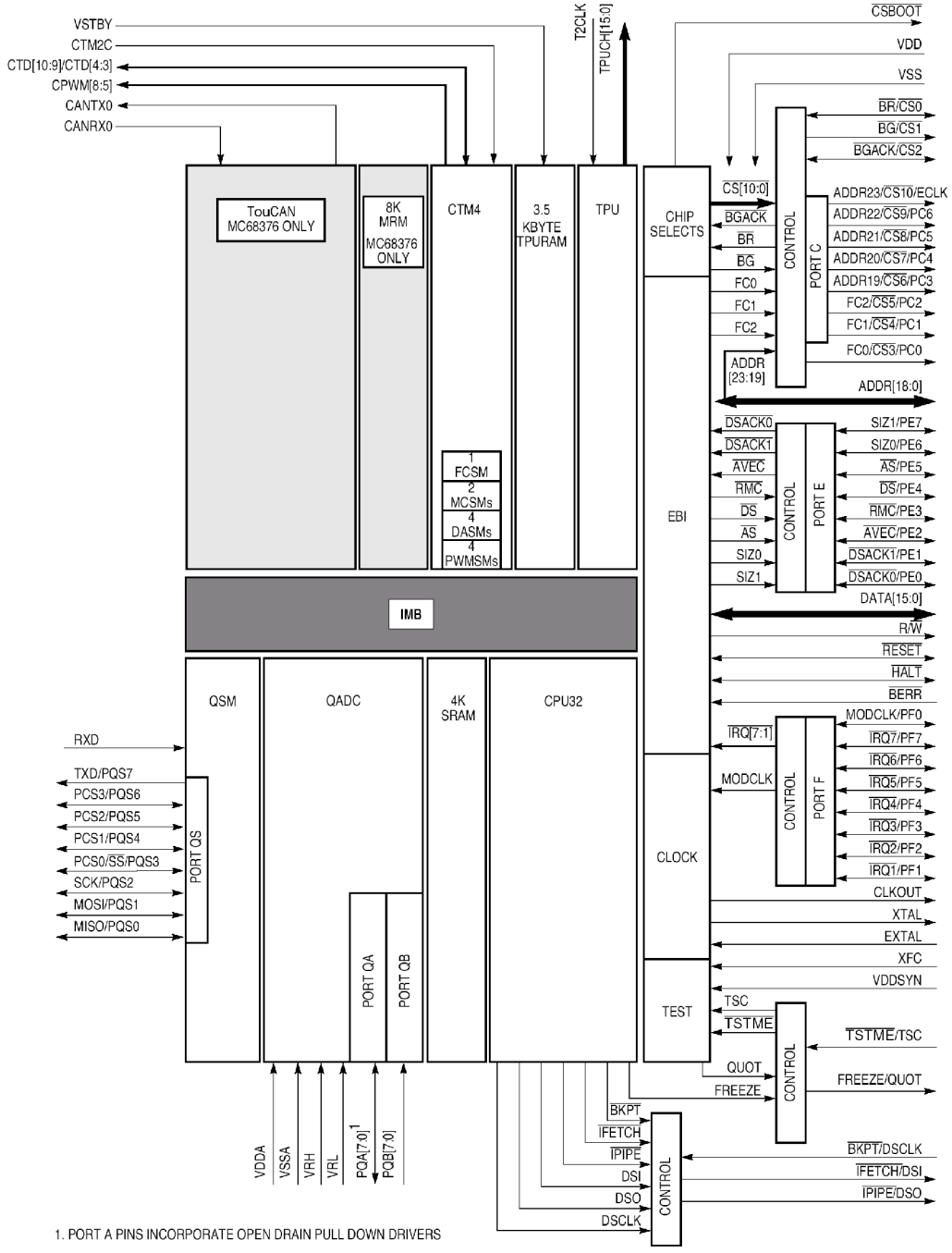
PIN-Count, Preis, Störunempfindlichkeit, Anpaßbarkeit

32-Bit Prozessoren Markanteile 1998 (Computer Zeitung 16/99)

Motorola 68K/Coldfire	34,1%
Mips	20,5%
ARM	19,6%
Hitachi Super-H	10,6%
Intel/AMD/... X86	5,0%
Intel i960	3,7%
IBM/Motorola PPC	2,0%
AMD 29K	0,7%
Sonstige	0,9%

Micro-Controller Markanteile 1999 (wysiwig://66/http://internet.about.com/cs/micrcontrollers)

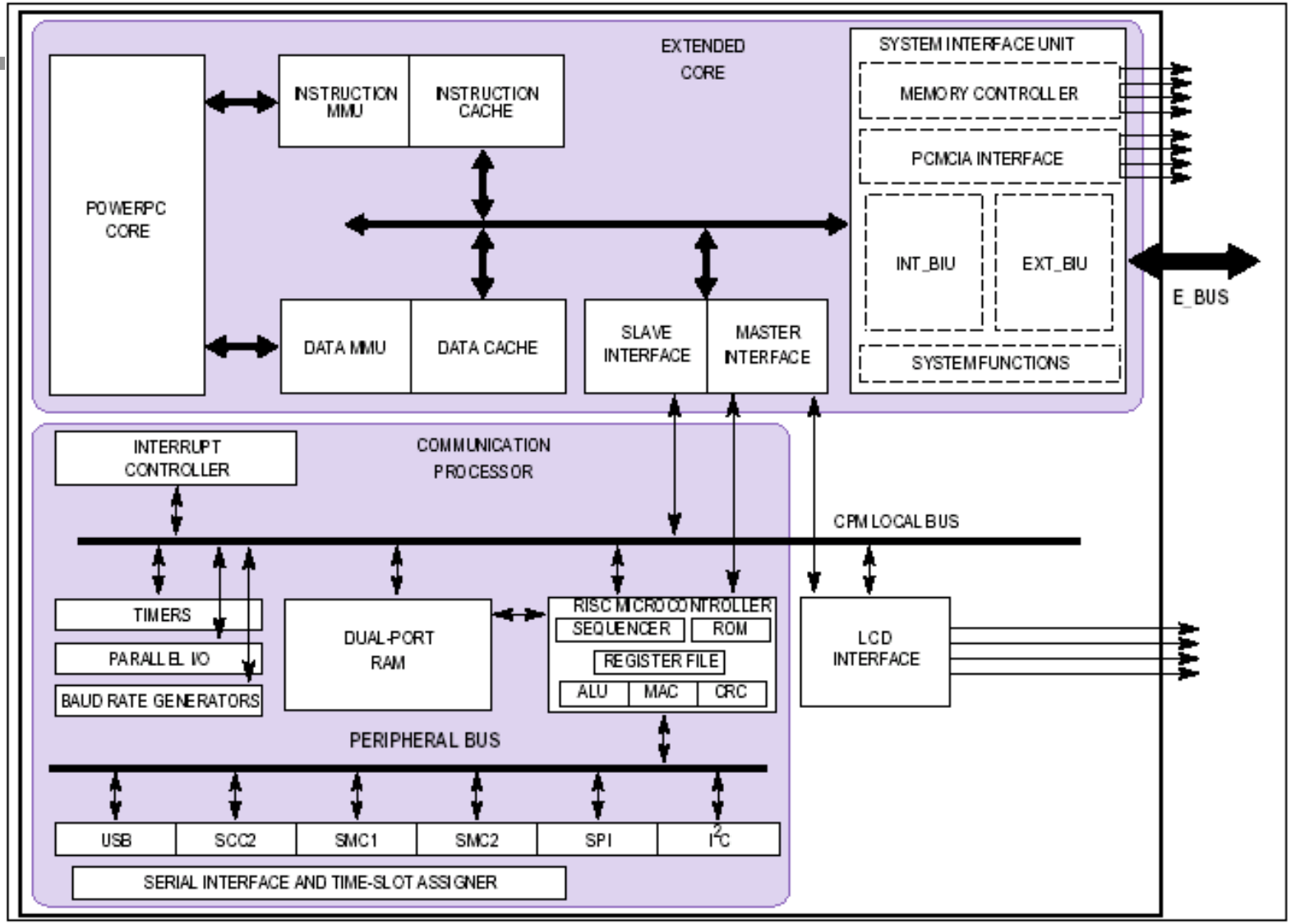
	Umsatz 10^9 $	Anteil %	Wachst.98-99 %
Motorola	2.54	18.0	25,1
Texas Instruments	2.47	17,5	35,0
Hitachi	1.28	9,1	- 3,8
Lucent	1.16	8,2	18,4
NEC	1.16	8,2	12,6
Mitsubishi	1.00	7,0	25,0
Intel	0,86	6,1	8,9
Philips	0,51	3,6	- 7,3



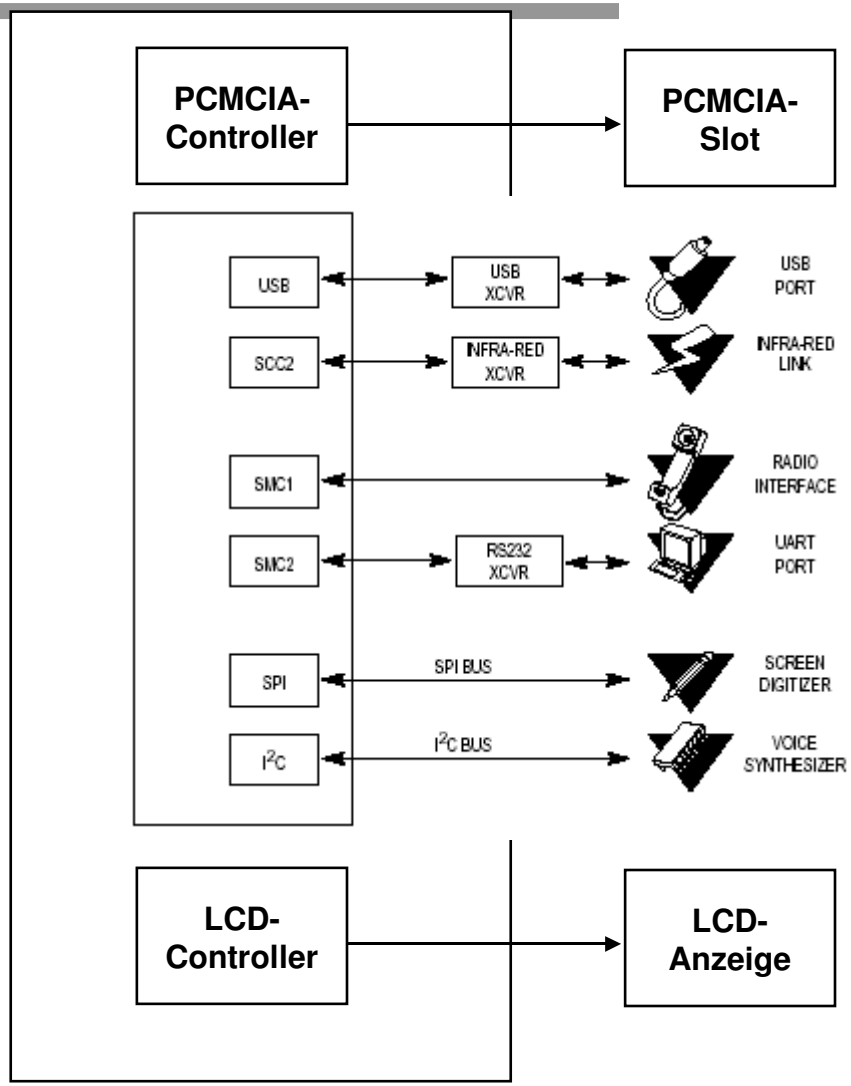
Blockschaltbild: MC 68376

IMB: Inter Module Bus
CTM: Config. Timer Module
QSM: Queued Serial Module
TPU: Time Processing Module
QADC: Queued ADC
EBI: Extended Bus Interface
TouCAN: CAN-Bus 2.0
MRM: Masked ROM Module

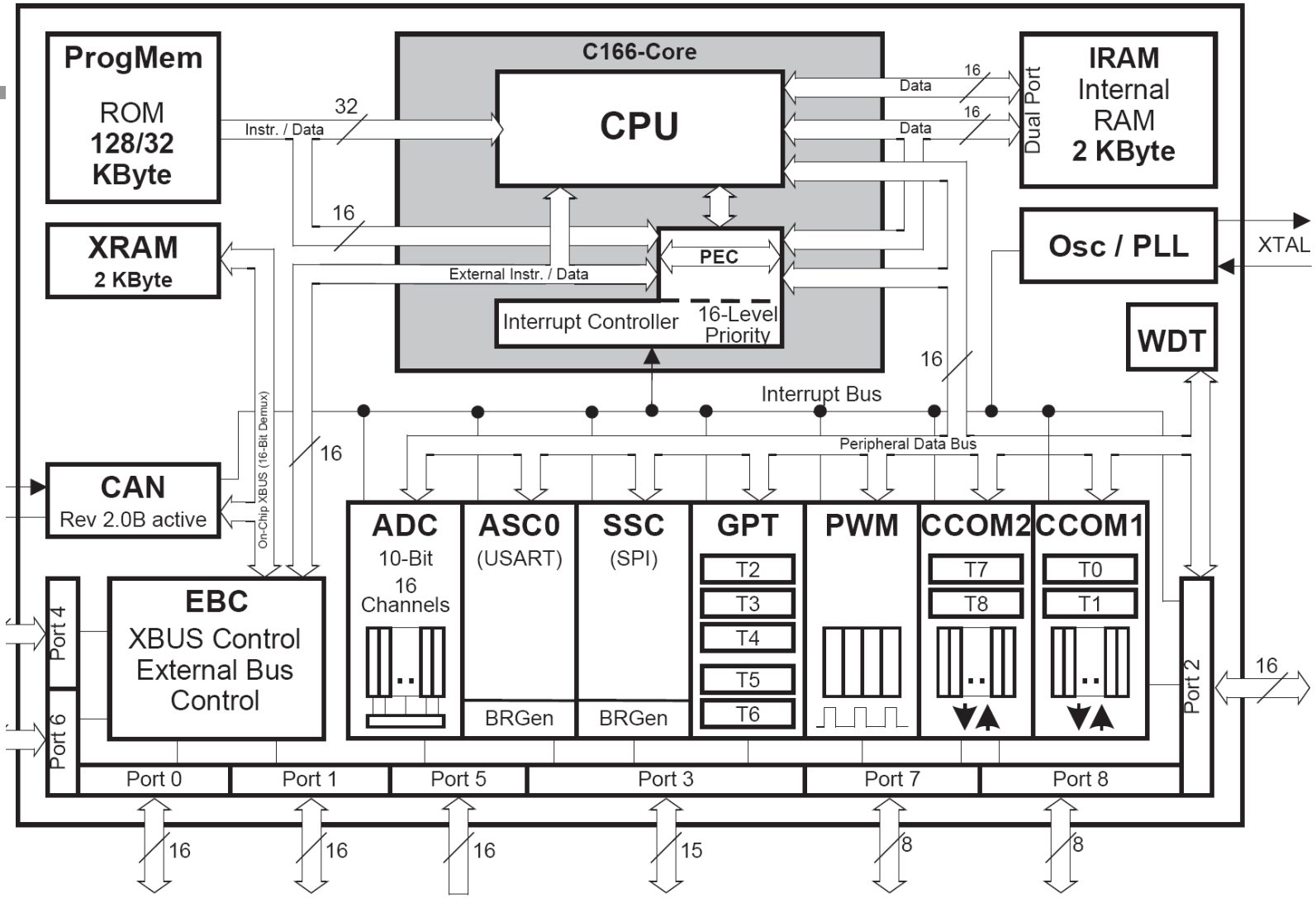
Power PC 823 embedded controller



Beispiel für einen PDA



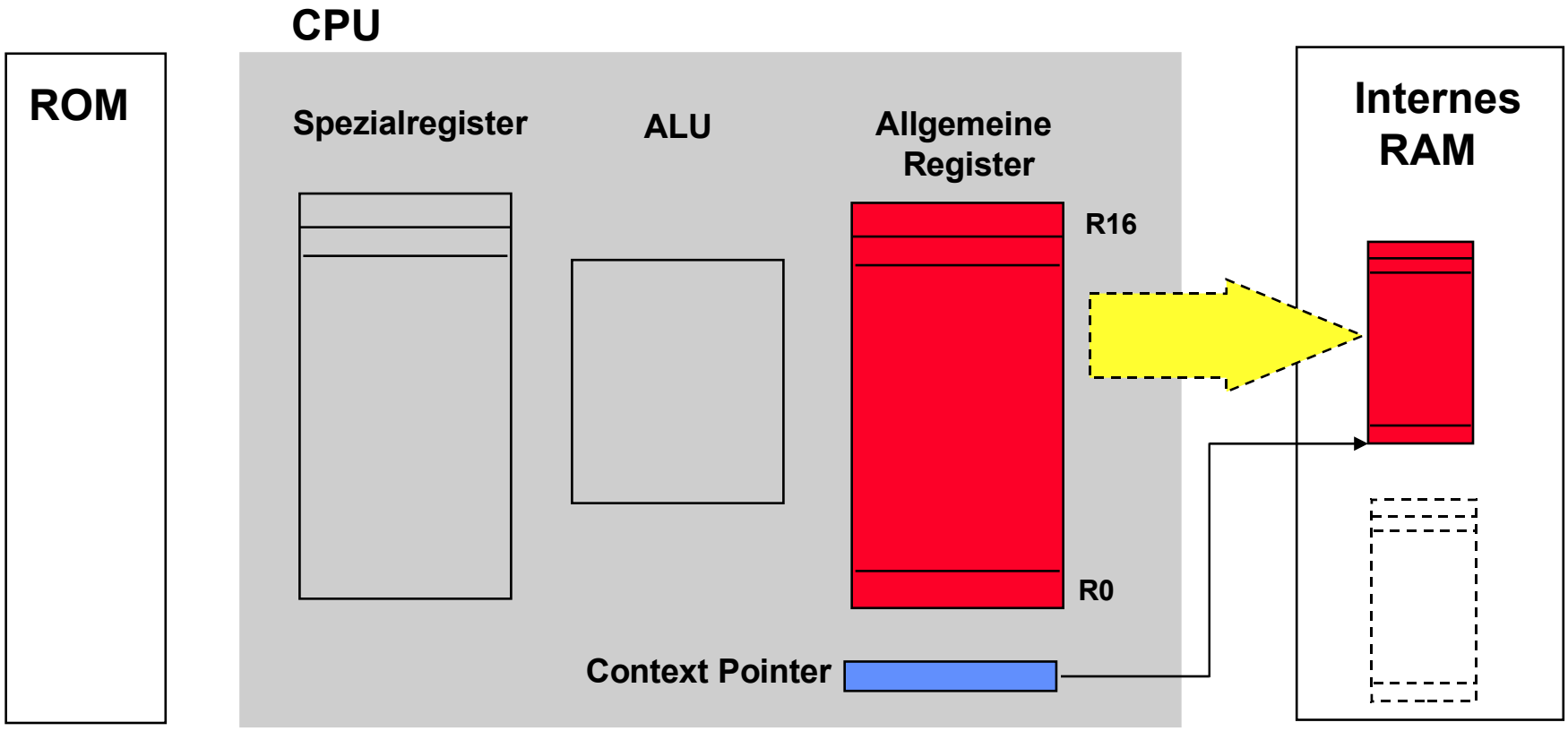
Infinion CR167



Infinion CR167

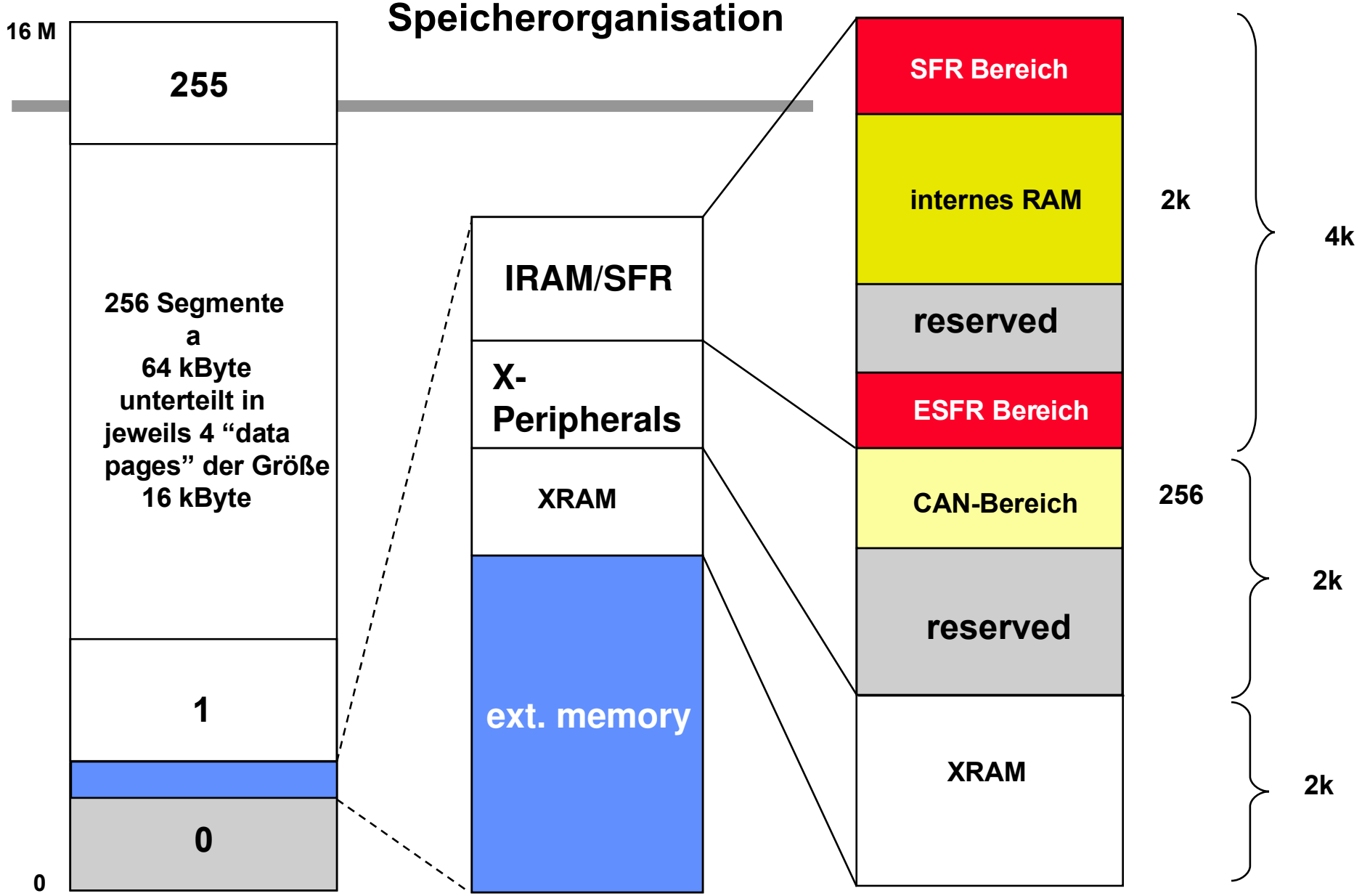
Organisation des interne RAM

Blockdiagramm der CPU C166/167



(Context Switch 100ns)

Speicherorganisation



Interner RAM Bereich und Bereich für die Special Functional Registers (SFRs)

Internes RAM wird genutzt:

- **für System Stack**
- **für allgemeine Register (mehrere Bänke)**
- **für den PEC (Peripheral Event Controller)**
- **als RAM**

SFRs:

kontrollieren die gesamten on-chip Spezialfunktionen wie:

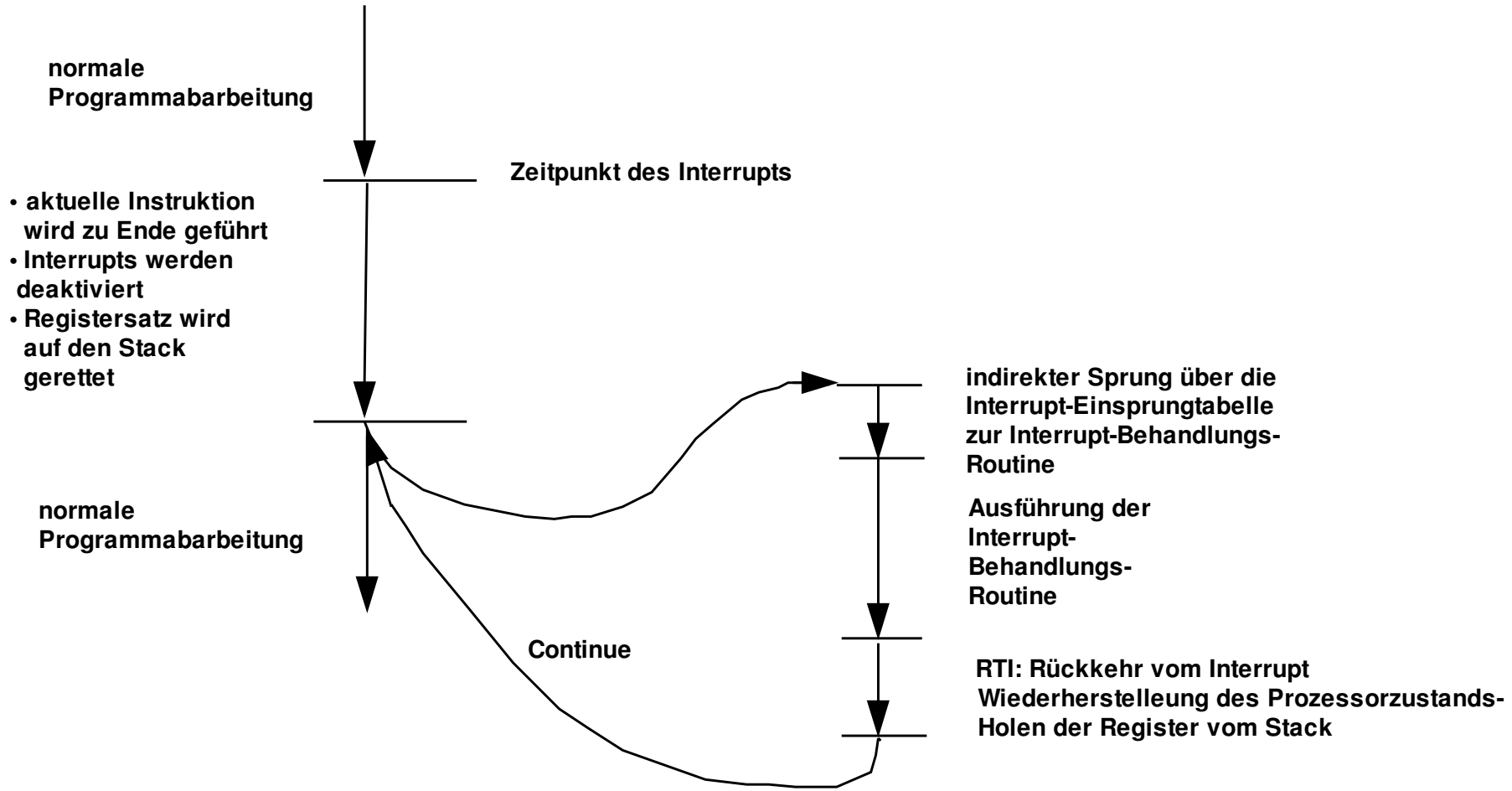
- **Ein-Ausgabe,**
- **A/D-Wandler,**
- **Kommunikation (CAN)**
- **Timer**

Interruptverarbeitung

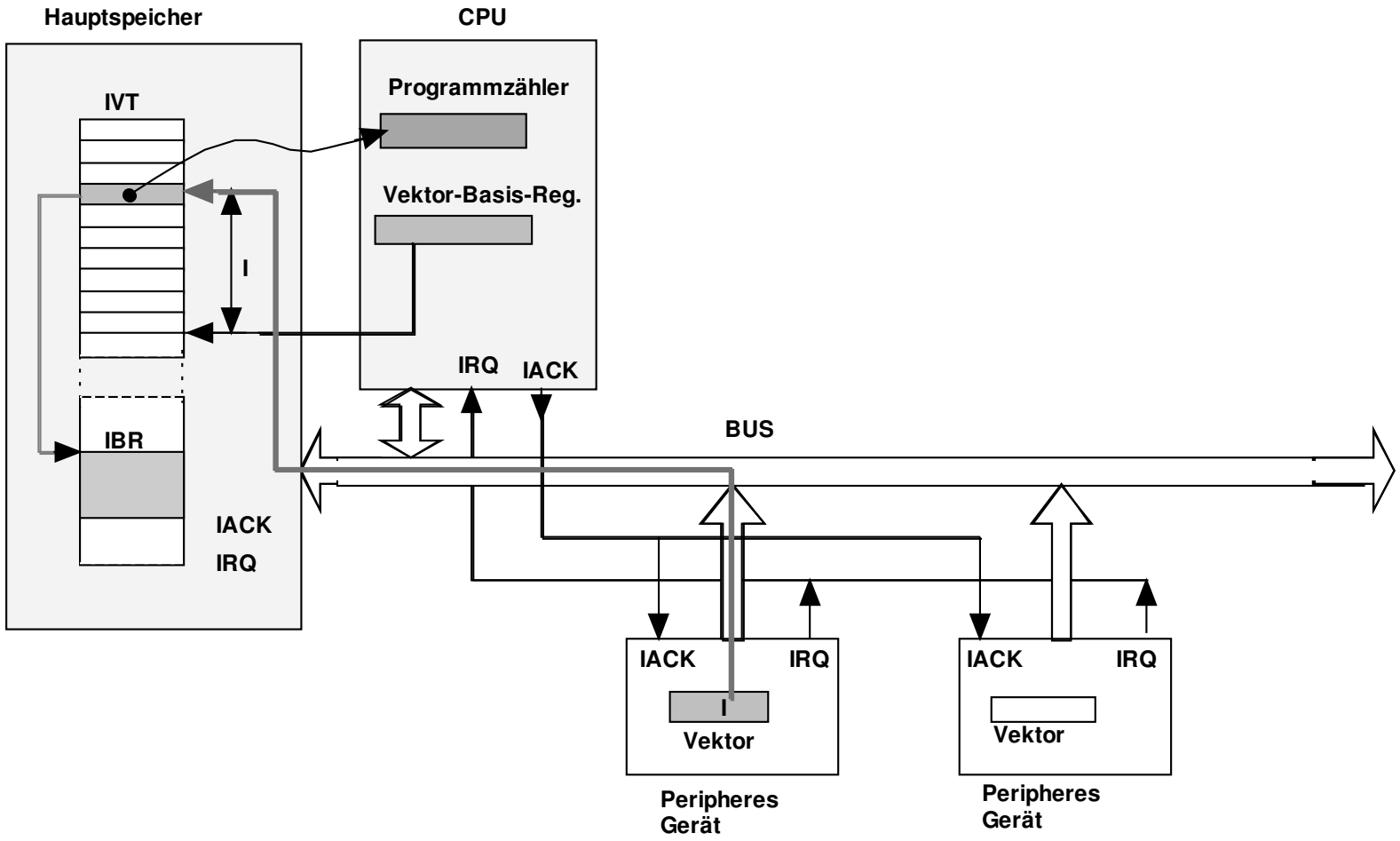
- **Reaktion auf externe Ereignisse**
- **Reaktion auf interne Funktionseinheiten, z.B.**
  - **ADC**
  - **Timer**
  - **Kommunikationseinheiten**

Ziele: **Nebenläufige Bearbeitung verschiedener Vorgänge**
Minimale Beanspruchung der CPU

Abarbeitung eines Interrupts



Vektorisierte Unterbrechungsbearbeitung



Interrupt und Trap Funktionen im C167

Normale Interrupt-Verarbeitung

- Interrupts von On-Chip Komponenten

Interrupt Verarbeitung über den PEC (Peripheral Event Controller)

- Minimale Interferenz mit der CPU (nur 1 Befehlszyklus)
- Kein Abspeichern des CPU-Status notwendig

Trap - Funktionen

- Software Traps
- Hardware Traps ausgelöst durch Befehlesausführung (z.B. illegal Opcode, Overflow, etc.)

Externe Interrupts

- Capture Input / Compare Output Leitungen
- Timer Input
- Fast External Interrupts (werden alle 50 ns gesampled, normal 400 ns bei einer 20 MHz Taktrate)

Source of Interrupt or PEC Service Request	Request Flag	Enable Flag	Interrupt Vector	Vector Location	Trap Number
CAPCOM Register 0	CC0IR	CC0IE	CC0INT	$00'0040_H$	$10_H / 16_D$
CAPCOM Register 1	CC1IR	CC1IE	CC1INT	$00'0044_H$	$11_H / 17_D$
CAPCOM Register 2	CC2IR	CC2IE	CC2INT	$00'0048_H$	$12_H / 18_D$
CAPCOM Register 3	CC3IR	CC3IE	CC3INT	$00'004C_H$	$13_H / 19_D$
CAPCOM Register 4	CC4IR	CC4IE	CC4INT	$00'0050_H$	$14_H / 20_D$
CAPCOM Register 5	CC5IR	CC5IE	CC5INT	$00'0054_H$	$15_H / 21_D$
CAPCOM Register 6	CC6IR	CC6IE	CC6INT	$00'0058_H$	$16_H / 22_D$
CAPCOM Register 7	CC7IR	CC7IE	CC7INT	$00'005C_H$	$17_H / 23_D$
CAPCOM Register 8	CC8IR	CC8IE	CC8INT	$00'0060_H$	$18_H / 24_D$
CAPCOM Register 9	CC9IR	CC9IE	CC9INT	$00'0064_H$	$19_H / 25_D$
CAPCOM Register 10	CC10IR	CC10IE	CC10INT	$00'0068_H$	$1A_H / 26_D$
CAPCOM Register 11	CC11IR	CC11IE	CC11INT	$00'006C_H$	$1B_H / 27_D$
CAPCOM Register 12	CC12IR	CC12IE	CC12INT	$00'0070_H$	$1C_H / 28_D$
CAPCOM Register 13	CC13IR	CC13IE	CC13INT	$00'0074_H$	$1D_H / 29_D$
CAPCOM Register 14	CC14IR	CC14IE	CC14INT	$00'0078_H$	$1E_H / 30_D$
CAPCOM Register 15	CC15IR	CC15IE	CC15INT	$00'007C_H$	$1F_H / 31_D$
CAPCOM Register 16	CC16IR	CC16IE	CC16INT	$00'00C0_H$	$30_H / 48_D$
CAPCOM Register 17	CC17IR	CC17IE	CC17INT	$00'00C4_H$	$31_H / 49_D$
CAPCOM Register 18	CC18IR	CC18IE	CC18INT	$00'00C8_H$	$32_H / 50_D$
CAPCOM Register 19	CC19IR	CC19IE	CC19INT	$00'00CC_H$	$33_H / 51_D$
CAPCOM Register 20	CC20IR	CC20IE	CC20INT	$00'00D0_H$	$34_H / 52_D$
CAPCOM Register 21	CC21IR	CC21IE	CC21INT	$00'00D4_H$	$35_H / 53_D$
CAPCOM Register 22	CC22IR	CC22IE	CC22INT	$00'00D8_H$	$36_H / 54_D$
CAPCOM Register 23	CC23IR	CC23IE	CC23INT	$00'00DC_H$	$37_H / 55_D$
CAPCOM Register 24	CC24IR	CC24IE	CC24INT	$00'00E0_H$	$38_H / 56_D$
CAPCOM Register 25	CC25IR	CC25IE	CC25INT	$00'00E4_H$	$39_H / 57_D$
CAPCOM Register 26	CC26IR	CC26IE	CC26INT	$00'00E8_H$	$3A_H / 58_D$
CAPCOM Register 27	CC27IR	CC27IE	CC27INT	$00'00EC_H$	$3B_H / 59_D$
CAPCOM Register 28	CC28IR	CC28IE	CC28INT	$00'00F0_H$	$3C_H / 60_D$
CAPCOM Register 29	CC29IR	CC29IE	CC29INT	$00'0110_H$	$44_H / 68_D$
CAPCOM Register 30	CC30IR	CC30IE	CC30INT	$00'0114_H$	$45_H / 69_D$
CAPCOM Register 31	CC31IR	CC31IE	CC31INT	$00'0118_H$	$46_H / 70_D$

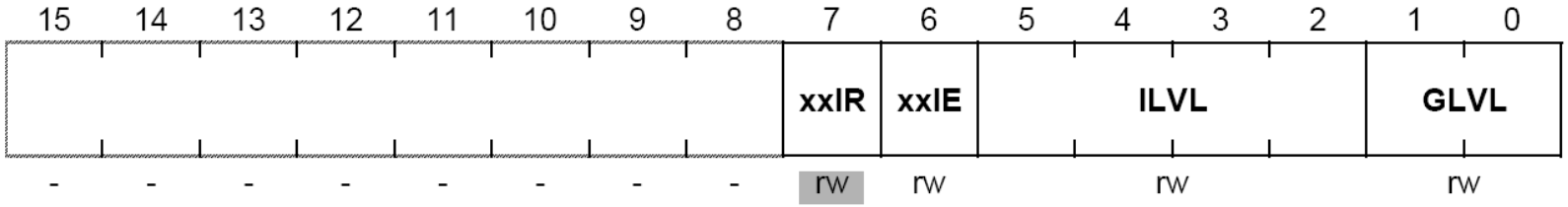
Source of Interrupt or PEC Service Request	Request Flag	Enable Flag	Interrupt Vector	Vector Location	Trap Number
CAPCOM Timer 0	T0IR	T0IE	T0INT	$00'0080_H$	$20_H / 32_D$
CAPCOM Timer 1	T1IR	T1IE	T1INT	$00'0084_H$	$21_H / 33_D$
CAPCOM Timer 7	T7IR	T7IE	T7INT	$00'00F4_H$	$3D_H / 61_D$
CAPCOM Timer 8	T8IR	T8IE	T8INT	$00'00F8_H$	$3E_H / 62_D$
GPT1 Timer 2	T2IR	T2IE	T2INT	$00'0088_H$	$22_H / 34_D$
GPT1 Timer 3	T3IR	T3IE	T3INT	$00'008C_H$	$23_H / 35_D$
GPT1 Timer 4	T4IR	T4IE	T4INT	$00'0090_H$	$24_H / 36_D$
GPT2 Timer 5	T5IR	T5IE	T5INT	$00'0094_H$	$25_H / 37_D$
GPT2 Timer 6	T6IR	T6IE	T6INT	$00'0098_H$	$26_H / 38_D$
GPT2 CAPREL Register	CRIR	CRIE	CRINT	$00'009C_H$	$27_H / 39_D$
A/D Conversion Complete	ADCIR	ADCIE	ADCINT	$00'00A0_H$	$28_H / 40_D$
A/D Overrun Error	ADEIR	ADEIE	ADEINT	$00'00A4_H$	$29_H / 41_D$
ASC0 Transmit	S0TIR	S0TIE	S0TINT	$00'00A8_H$	$2A_H / 42_D$
ASC0 Transmit Buffer	S0TBIR	S0TBIE	S0TBINT	$00'011C_H$	$47_H / 71_D$
ASC0 Receive	S0RIR	S0RIE	S0RINT	$00'00AC_H$	$2B_H / 43_D$
ASC0 Error	S0EIR	S0EIE	S0EINT	$00'00B0_H$	$2C_H / 44_D$
SSC Transmit	SSCTIR	SSCTIE	SSCTINT	$00'00B4_H$	$2D_H / 45_D$
SSC Receive	SSCRIR	SSCRIE	SSCRINT	$00'00B8_H$	$2E_H / 46_D$
SSC Error	SSCEIR	SSCEIE	SSCEINT	$00'00BC_H$	$2F_H / 47_D$
PWM Channel 0...3	PWMIR	PWMIE	PWMINT	$00'00FC_H$	$3F_H / 63_D$
CAN Interface	XP0IR	XP0IE	XP0INT	$00'0100_H$	$40_H / 64_D$
X-Peripheral Node 1	XP1IR	XP1IE	XP1INT	$00'0104_H$	$41_H / 65_D$
X-Peripheral Node 2	XP2IR	XP2IE	XP2INT	$00'0108_H$	$42_H / 66_D$
PLL Unlock	XP3IR	XP3IE	XP3INT	$00'010C_H$	$43_H / 67_D$

Note: Each entry of the interrupt vector table provides room for two word instructions or one doubleword instruction. The respective vector location results from multiplying the trap number by 4 (4 bytes per entry).
For devices which do not incorporate a CAN Module or a PLL the respective interrupt nodes may be used for software triggered interrupts (see X-Peripheral node n).

Interrupts und Traps des C 167

Interrupt Kontrolle: Einteilung in Gruppen und individuelle Prioritäten

xxIC ($yyyy_H$ / zz_H) <SFR area> **Reset Value: - - 00_H**



Bit	Function
GLVL	**Group Level** Defines the internal order for simultaneous requests of the same priority. 3: Highest group priority 0: Lowest group priority
ILVL	**Interrupt Priority Level** Defines the priority level for the arbitration of requests. F_H: Highest priority level 0_H: Lowest priority level
xxIE	**Interrupt Enable Control Bit** (individually enables/disables a specific source) '0': Interrupt request is disabled '1': Interrupt Request is enabled
xxIR	**Interrupt Request Flag** '0': No request pending '1': This source has raised an interrupt request

Beispiel für Gruppen- und individuelle Interruptprioritäten

Priority Level			Type of Service	
ILVL		GLVL	COUNT = 00H	COUNT ≠ 00_H
1 1 1	**1**	**1 1**	CPU interrupt, level 15, group priority 3	PEC service, channel 7
1 1 1	**1**	**1 0**	CPU interrupt, level 15, group priority 2	PEC service, channel 6
1 1 1	**0**	**1 0**	CPU interrupt, level 14, group priority 2	PEC service, channel 2
1 1 0	1	1 0	CPU interrupt, level 13, group priority 2	CPU interrupt, level 13, group priority 2
0 0 0	1	1 1	CPU interrupt, level 1, group priority 3	CPU interrupt, level 1, group priority 3
0 0 0	1	0 0	CPU interrupt, level 1, group priority 0	CPU interrupt, level 1, group priority 0
0 0 0	0	X X	No service!	No service!

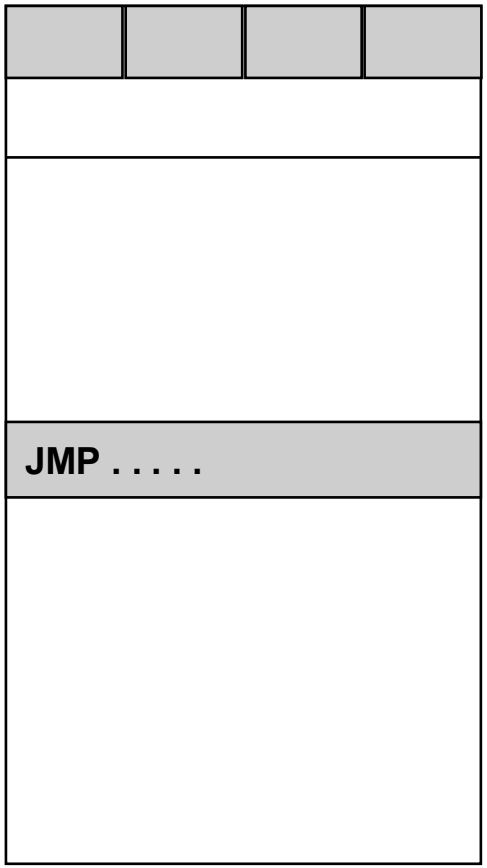
Interrupt Classes

* ILVL: Interrupt Level Vector List
** GLVL: Group Level Vector List

ILVL* (priority)	GLVL** 11	10	01	00	Interpretation
15					PEC Service on up to 8 channels
14					
13					
7	x	x	x	x	Interrupt Class 1
11	x				5 sources on 2 levels
10					
9					
8	x	x	x	x	Interrupt Class 2
7	x	x	x	x	9 sources on 3 levels
6	x				
5	x	x	x	x	Interrupt Class 3
4	x				5 sources on 2 levels
3					
2					
1					
0					no service

Fast Interrupt Handling

IVT
Interrupt Vector Table



512 Byte
128 x 4-Byte Einträge

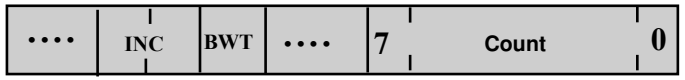
Bei Interrupt werden ein Befehl (4 Byte) oder 2 Befehle (2 Byte) ausgeführt. Normalerweise wird ein JMP zur Behandlungsroutine ausgeführt.

PEC: Peripheral Event Controller

- **Schnelle Alternative zur normalen Interruptverarbeitung**
- **Erlaubt den Transfer eines einzelnen Datums mit minimaler CPU-Belastung**
    - **CPU-Aktivität wird nur für einen einzigen Zyklus unterbrochen**
    - **kein interner Zustand muß gerettet werden**
    - **Prioritätsebenen 14 oder 15.**

Datentransfer mit PEC

PEC-Cntrl.-Reg

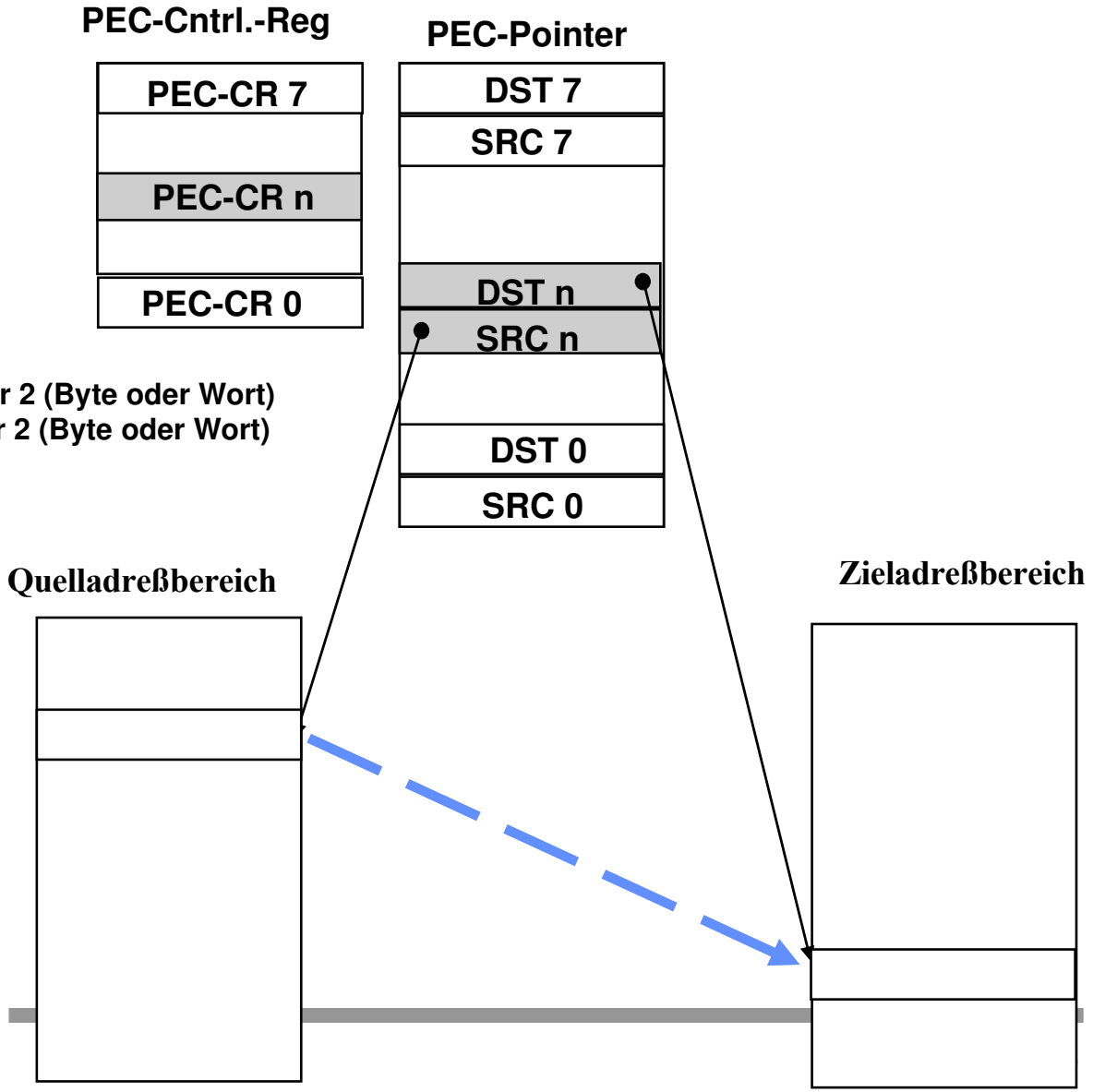


Feld:

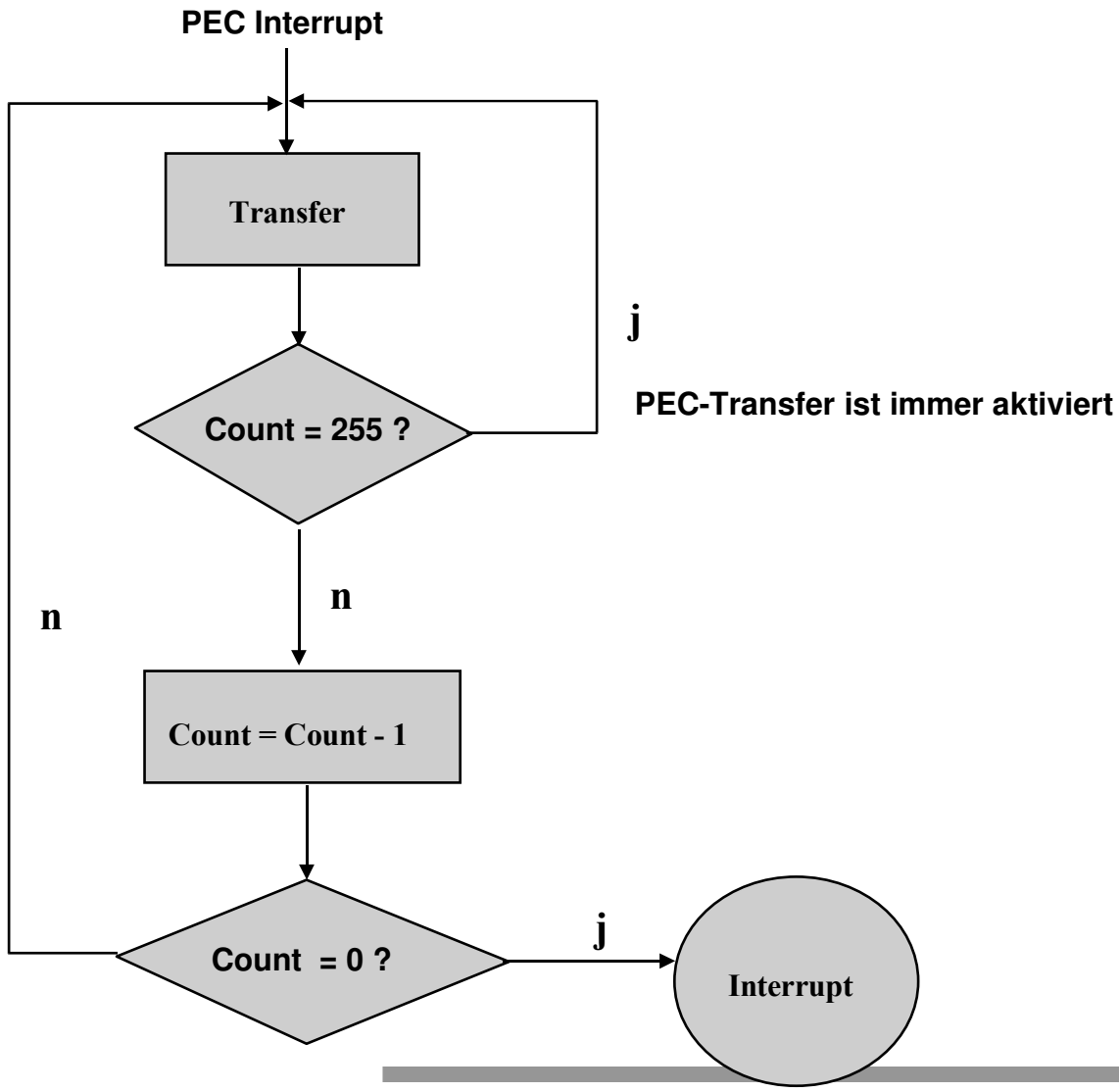
INC kontrolliert die SRC und DST PEC-Pointer
00: keine Modifikation
01: Incrementiere DSTx um 1 oder 2 (Byte oder Wort)
10: Incrementiere SRCx um 1 oder 2 (Byte oder Wort)
11: Reserviert

BWT: 0: Transferiere Wort
1: Transferiere Byte

Count: Anzahl der zu transferierenden Bytes bzw. Worte



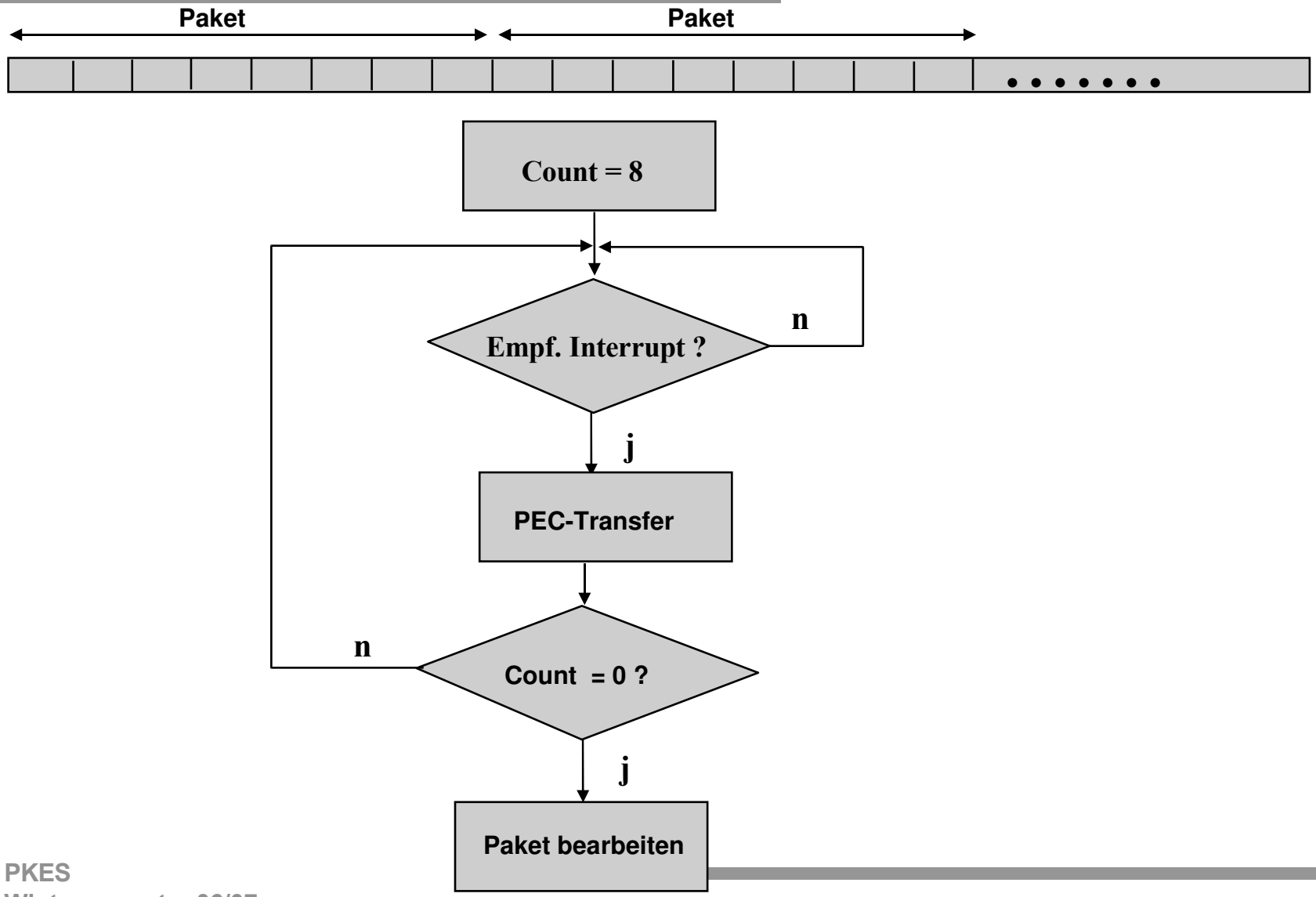
Steuerung des PEC- Datentransfers



PEC-Transfer ist immer aktiviert

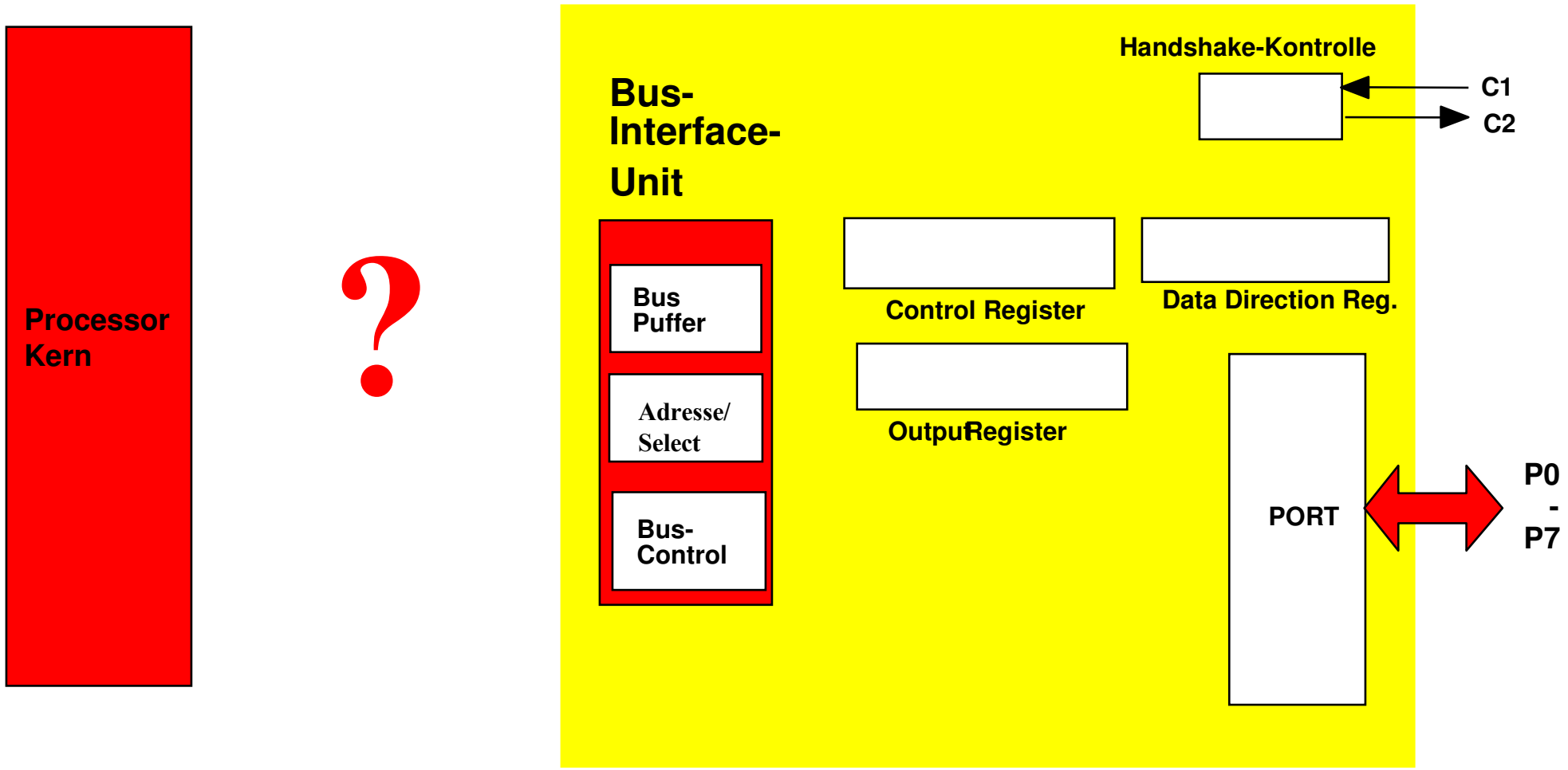
PEC-Anwendungsbeispiel: asynchrone Kommunikation

- jeweils 8 Byte-Pakete sollen empfangen werden.

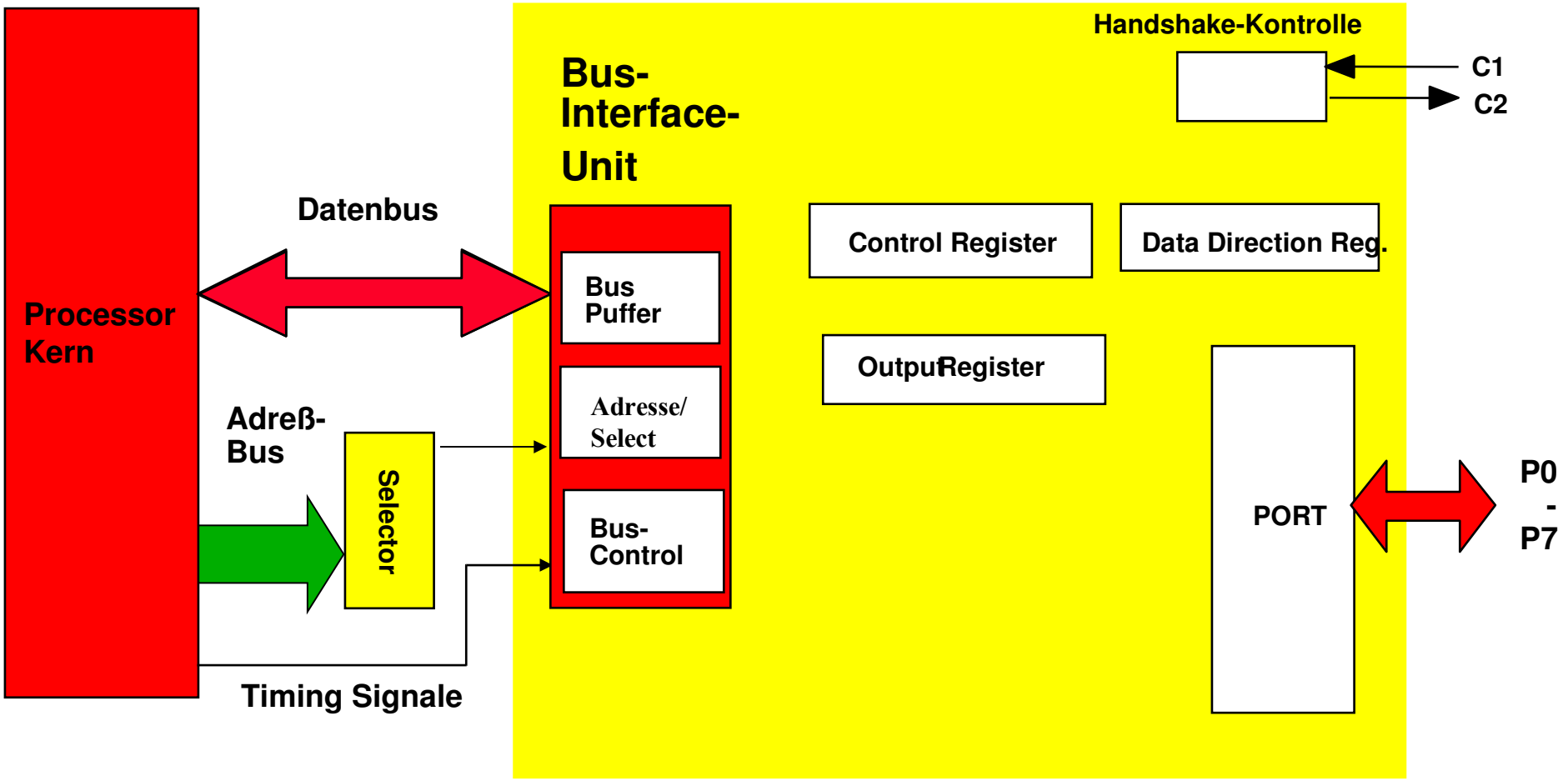


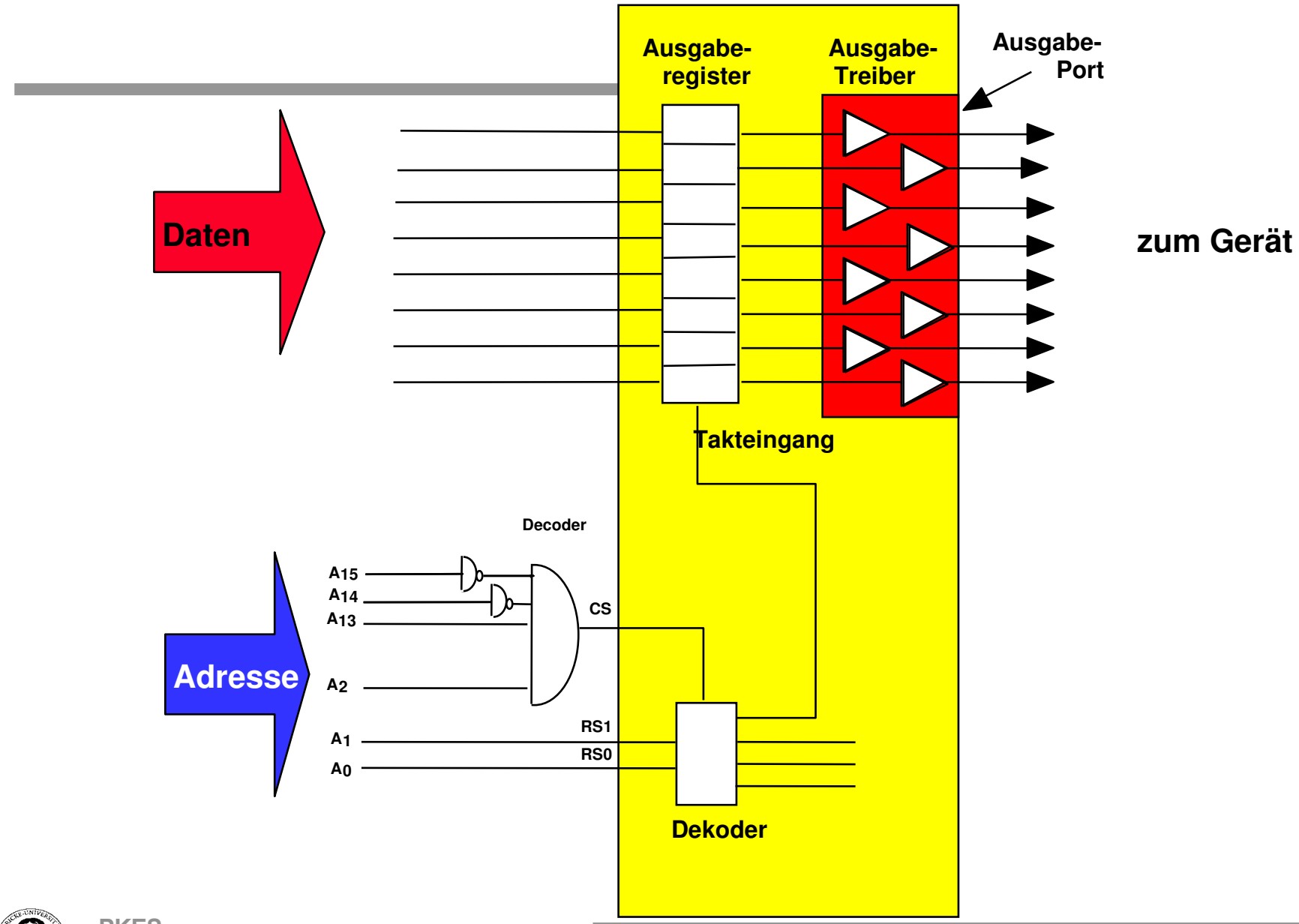
Ports, die Schnittstelle zur Peripherie

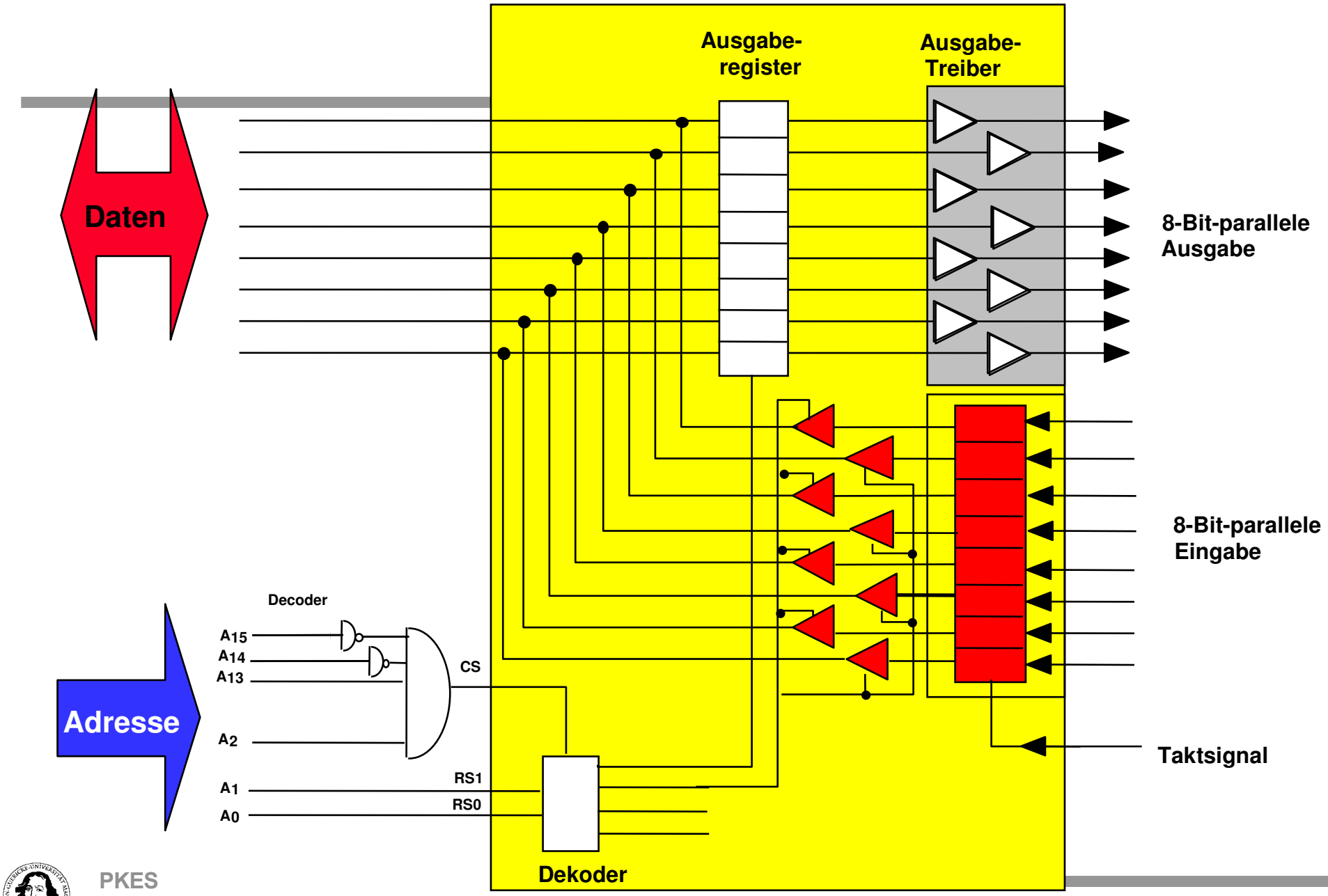
Eine Byte-parallele Geräteschnittstelle

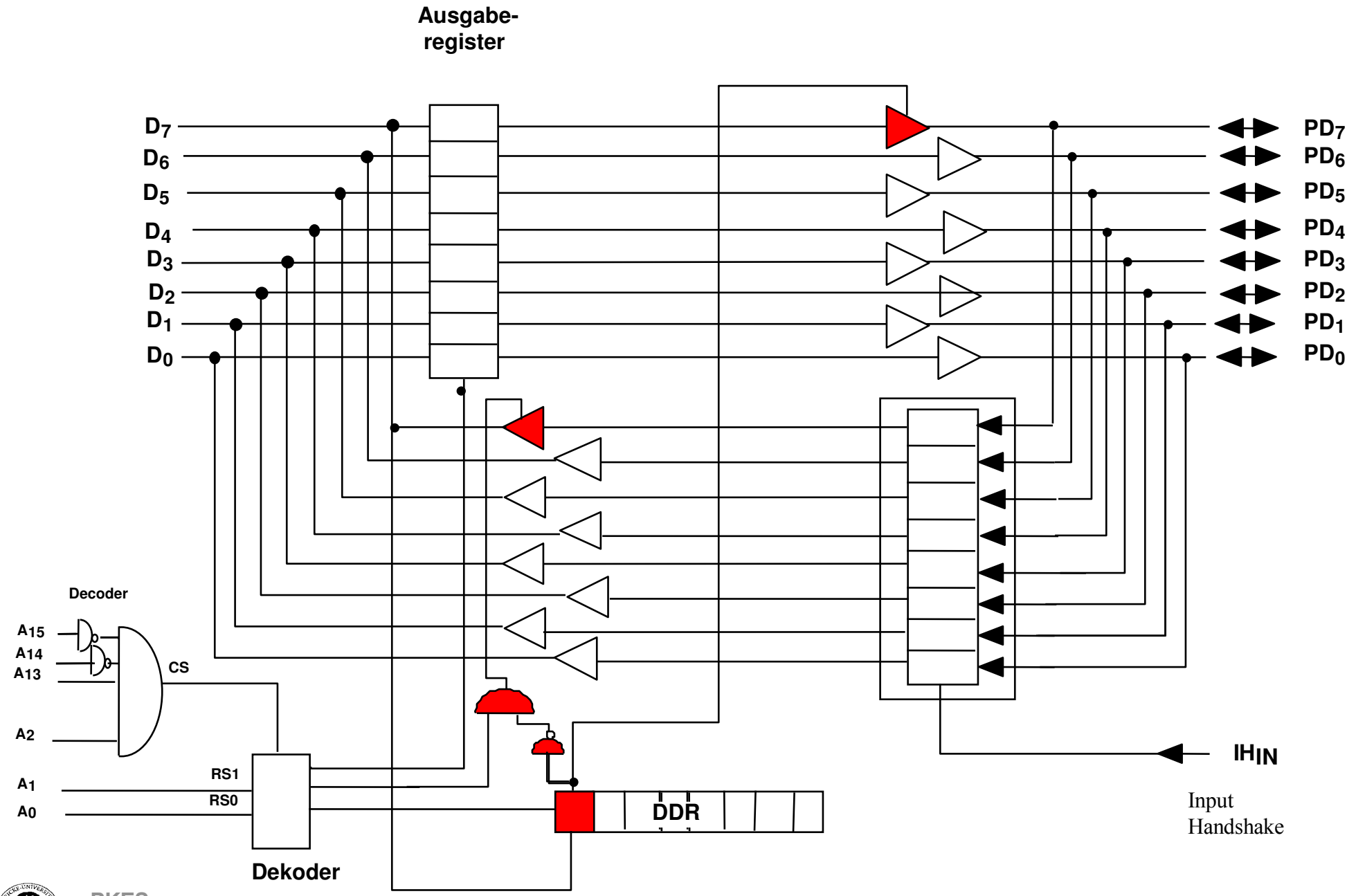


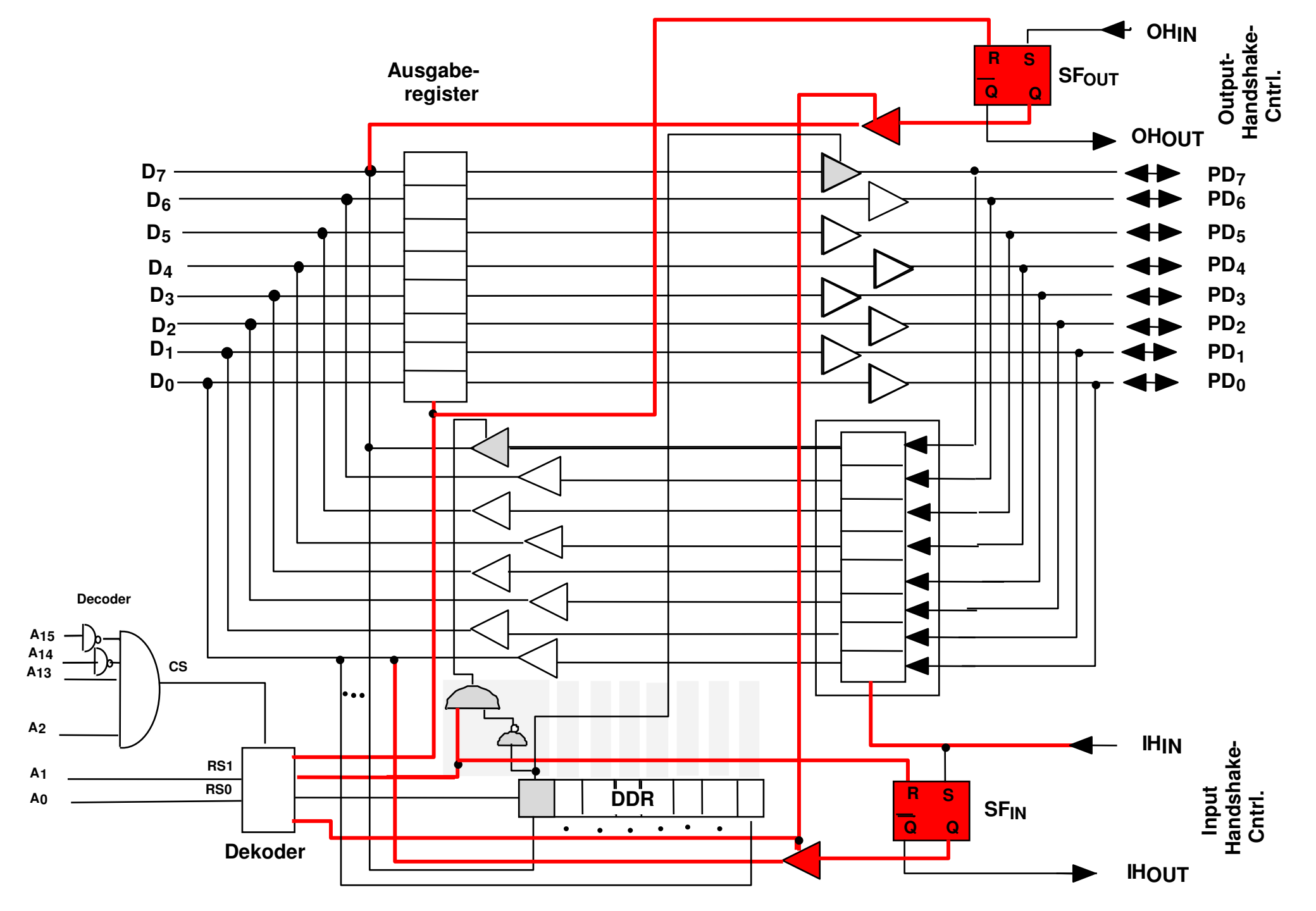
Eine Byte-parallele Geräteschnittstelle




Ausgabe-
register
Ausgabe-
Treiber
Ausgabe-
Port
Daten
zum Gerät
Takteingang
Decoder
A15
A14
A13
CS
Adresse
A2
RS1
A1
RS0
A0
Dekoder


Ausgabe-
register
Ausgabe-
Treiber
Daten
8-Bit-parallele
Ausgabe
8-Bit-parallele
Eingabe
Decoder
A15
A14
A13
CS
Adresse
A2
RS1
A1
RS0
A0
Taktsignal
Dekoder


Ausgabe-
register
D7
D6
D5
D4
D3
D2
D1
D0
PD7
PD6
PD5
PD4
PD3
PD2
PD1
PD0
Decoder
A15
A14
A13
CS
A2
A1
A0
RS1
RS0
Dekoder
DDR
IHIN
Input
Handshake


Ausgabe-
register
D7
D6
D5
D4
D3
D2
D1
D0
R
S
Q
Q
SFOUT
OHIN
OHOUT
Output-
Handshake-
Cntrl.
PD7
PD6
PD5
PD4
PD3
PD2
PD1
PD0
Decoder
A15
A14
A13
CS
A2
A1
A0
RS1
RS0
Dekoder
DDR
SFIN
IHIN
IHOUT
Input
Handshake-
Cntrl.

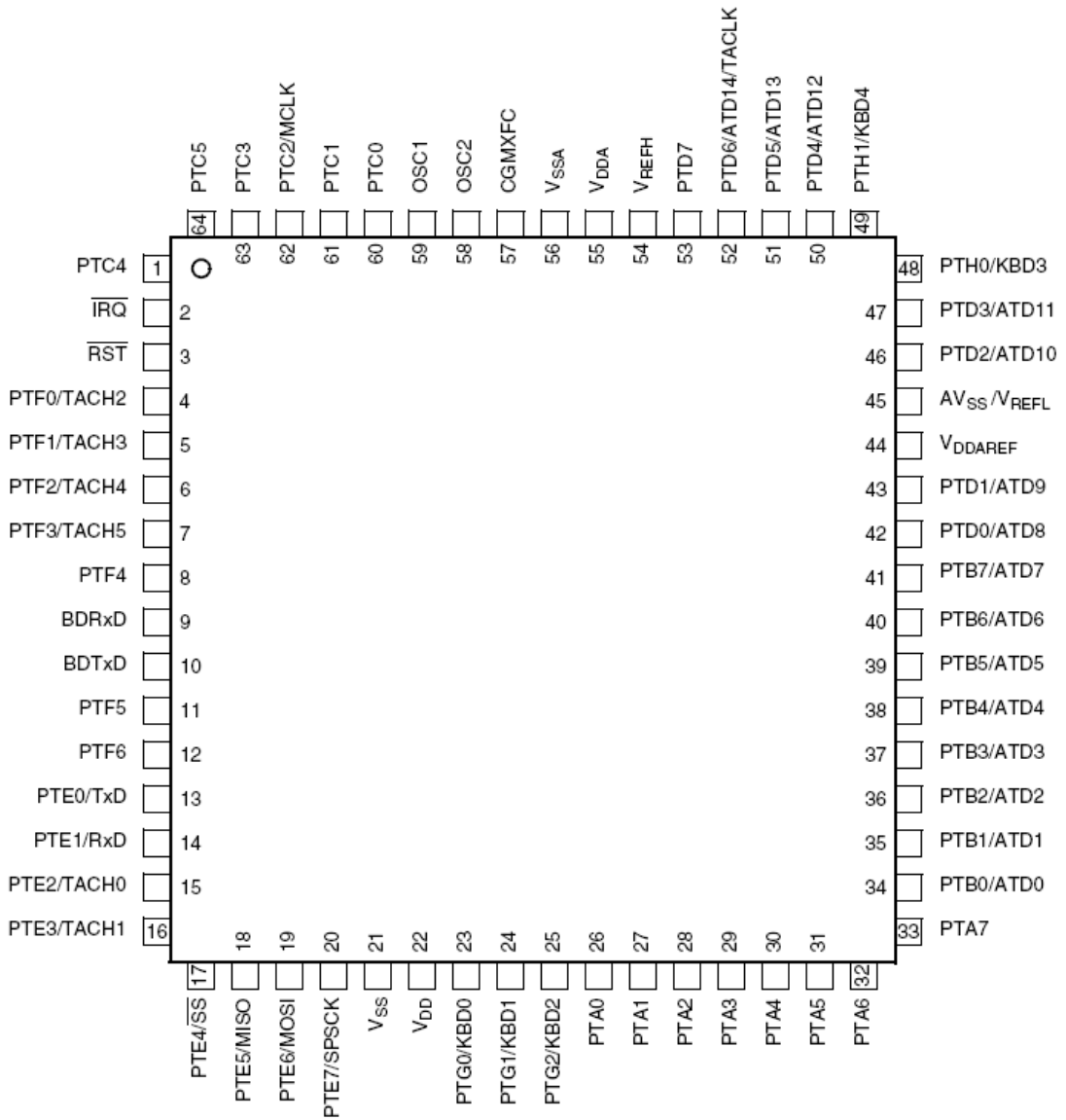
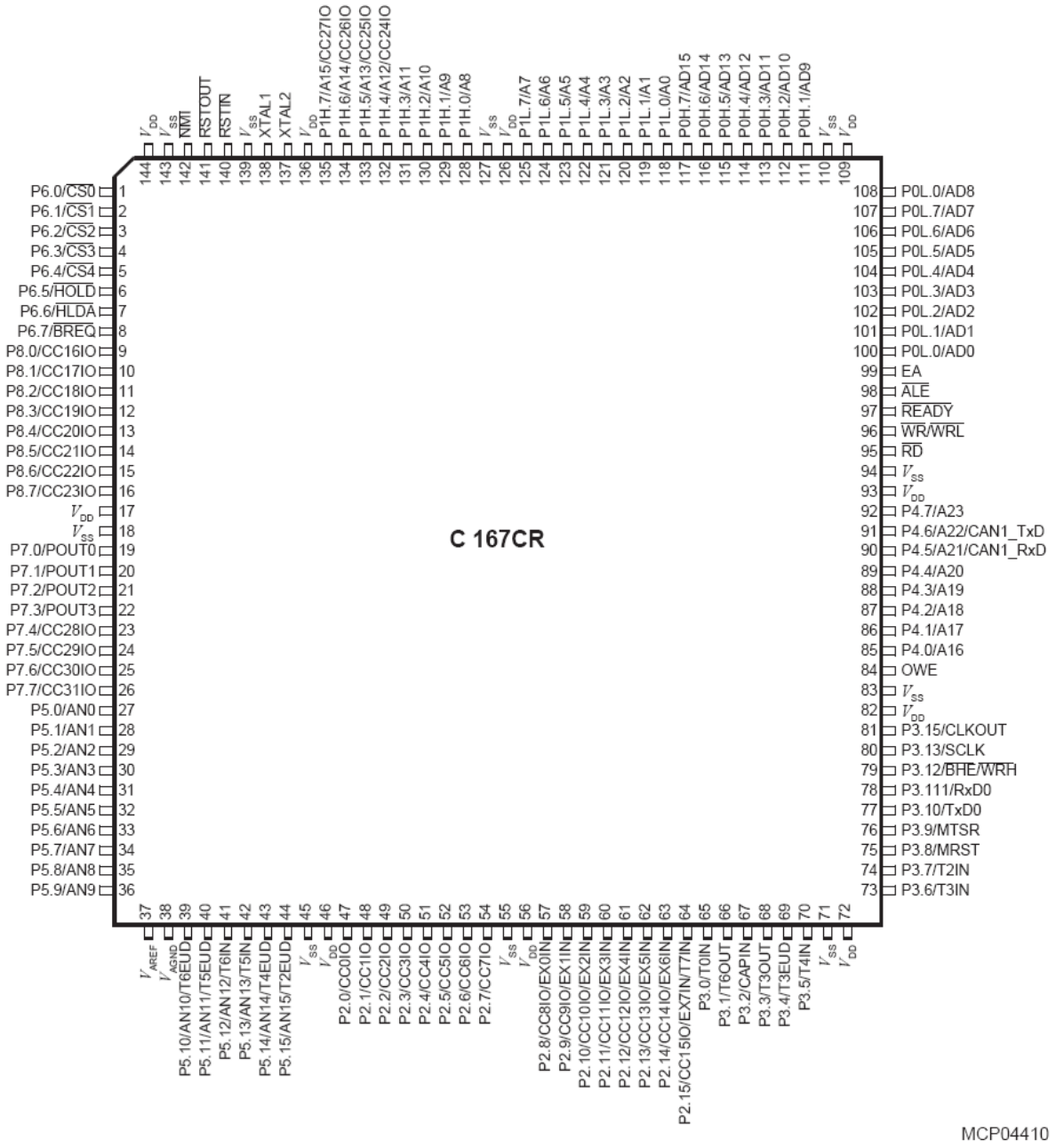


Figure 1-4. MC68HC908AS60A (64-Pin QFP)

C 167 Pins



MCP04410

Infineon C 167 I/O-Ports

111 I/O-Leitungen organisiert in:

1	**16-Bit I/O-Port**	**(Port 2)**
2	**2x8-Bit I/O-Ports**	**(Port 0 (P0H, P0L), Port 1 (P1H, P1L))**
4	**8-Bit I/O-Ports**	**(Port 4, 6, 7, 8)**
1	**15-Bit I/O-Port**	**(Port 3)**
1	**16-Bit Input Port**	**(Port 5, Analoge Eingänge)**

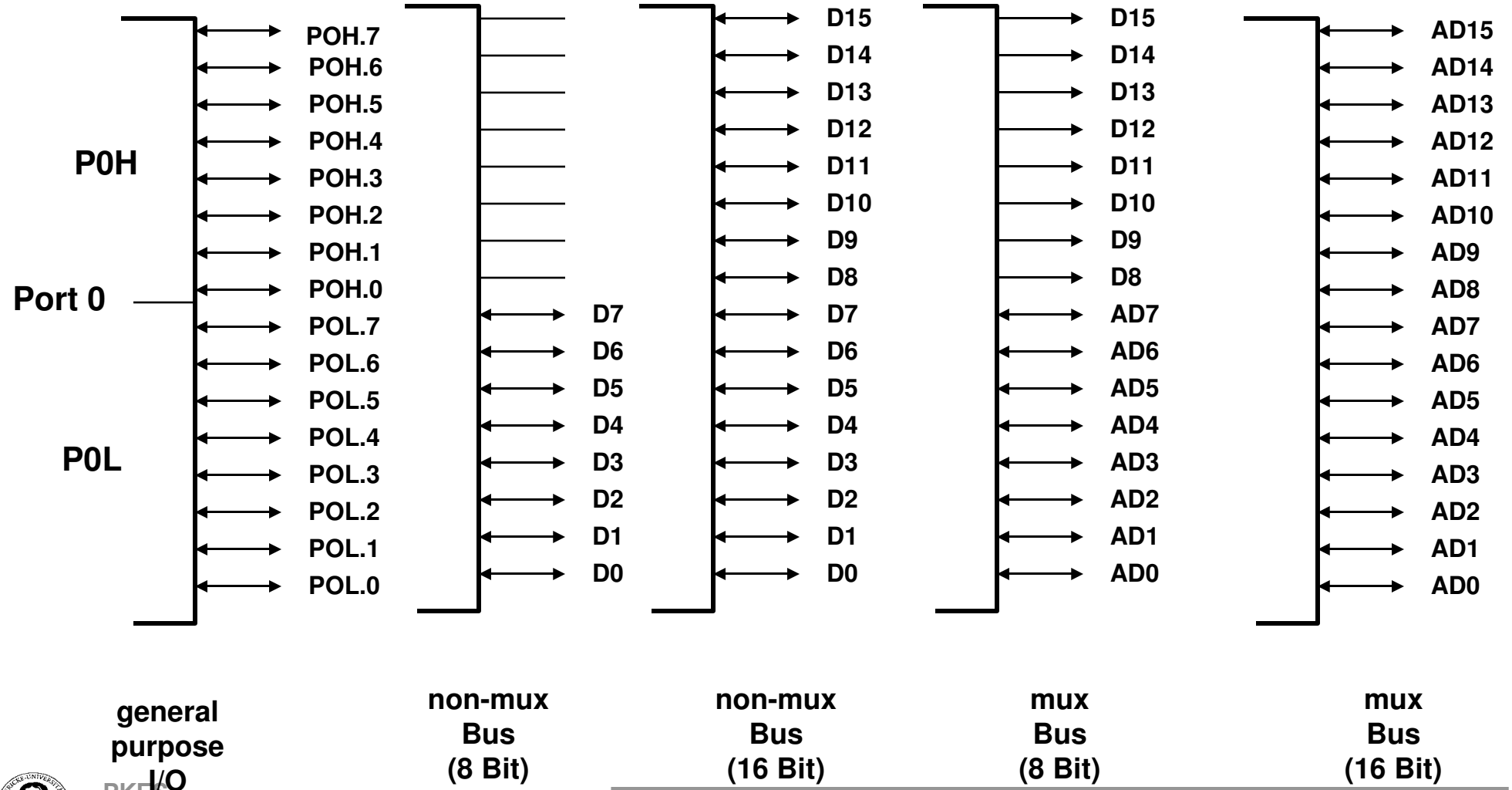
Ports können meist in einer Vielzahl von Modi sehr flexibel den Anforderungen angepaßt werden.

Bedeutung für Port-Pins für Port 0:

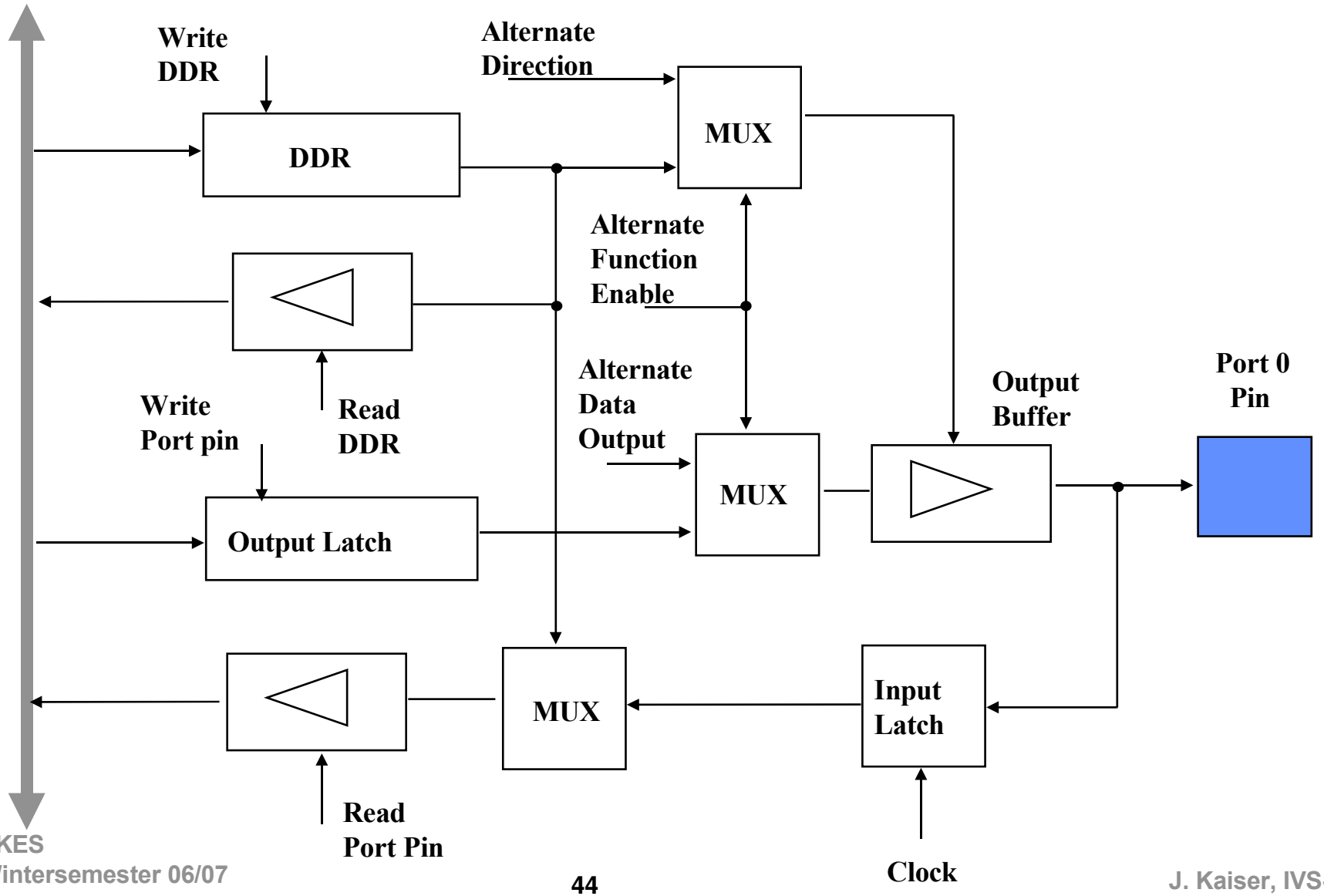
- General Purpose I/O-Pins
- 8-Bit Datenbus
- 16-Bit Adreßbus
- 8-Bit Daten-/16-Bit Adreßbus (multiplexed mode)
- 16-Bit Daten-/16-Bit Adreßbus (multiplexed mode)

Spezialfunktionen der übrigen Ports (Alternate Functions) :

Port 1: Capture & Compare
Port 2: Fast External Interrupt Inputs
Port 3: Timer Input & Output, Serial synch. & async. Communication Channels
Port 4: CAN TxD & RxD
Port 5: Analog In
Port 6: Chip Select Lines for Memory Extensions
Port 7: PWM Channels, Capture & Compare
Port 8: Capture & Compare


Alternate Function →
a.)
b.)
c.)
d.)
P0H
Port 0
P0L
POH.7
POH.6
POH.5
POH.4
POH.3
POH.2
POH.1
POH.0
POL.7
POL.6
POL.5
POL.4
POL.3
POL.2
POL.1
POL.0
D7
D6
D5
D4
D3
D2
D1
D0
D15
D14
D13
D12
D11
D10
D9
D8
D7
D6
D5
D4
D3
D2
D1
D0
D15
D14
D13
D12
D11
D10
D9
D8
AD7
AD6
AD5
AD4
AD3
AD2
AD1
AD0
AD15
AD14
AD13
AD12
AD11
AD10
AD9
AD8
AD7
AD6
AD5
AD4
AD3
AD2
AD1
AD0
general purpose I/O
non-mux Bus (8 Bit)
non-mux Bus (16 Bit)
mux Bus (8 Bit)
mux Bus (16 Bit)

Beispiel für die Schaltung eines Port Pins

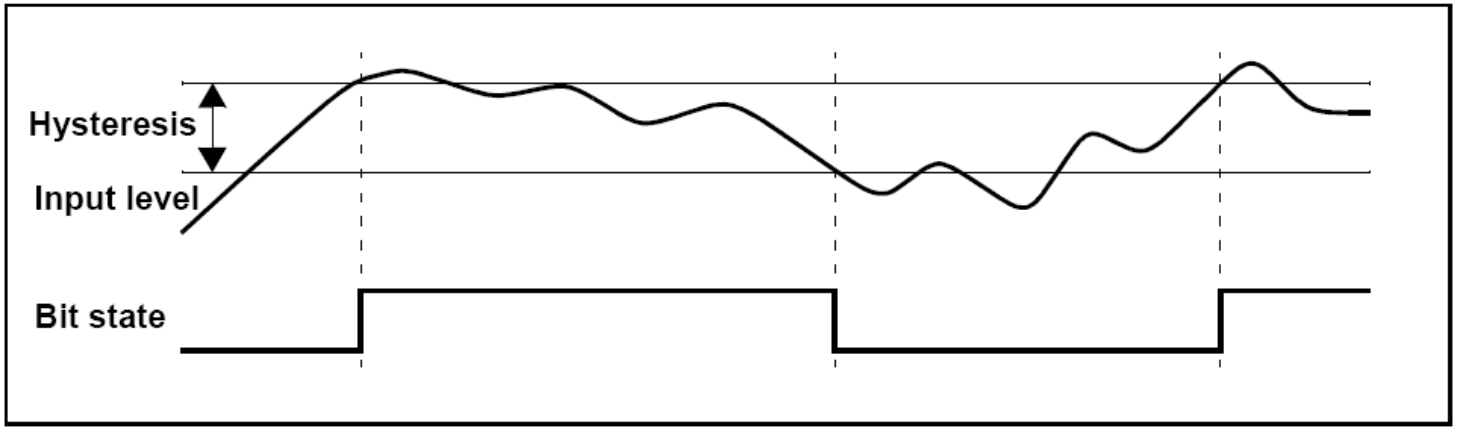


Festlegung der Eingangshysterese

Bit	Function
PxLIN	**Port x Low Byte Input Level Selection** 0 : Pins Px.7...Px.0 switch on standard TTL input levels 1 : Pins Px.7...Px.0 switch on special threshold input levels
PxHIN	**Port x High Byte Input Level Selection** 0 : Pins Px.15...Px.8 switch on standard TTL input levels 1 : Pins Px.15...Px.8 switch on special threshold input levels

All options for individual direction and output mode control are availbale for each pin independent from the selected input threshold.

The input hysteresis provides stable inputs from noisy or slowly changing external signals.



Bestimmung der Ausgangstreibercharakteristik